UNIVERSITAT POLITÈCNICA DE CATALUNYA

Doctoral Programme

AUTOMATIC CONTROL, ROBOTICS AND COMPUTER VISION

Ph.D. Thesis

FOUNDATIONS OF ONTOLOGY-BASED EXPLAINABLE ROBOTS

Alberto Olivares Alarcos

Advisors:
Dr. Guillem Alenyà Ribas
Dr. Sergi Foix Salmerón

Barcelona, December 2024

Foundations of ontology-based explainable robots

A thesis submitted to the Universitat Politècnica de Catalunya
to obtain the degree of Doctor of Philosophy

Doctoral programme:
Automatic Control, Robotics and Computer Vision

This thesis was completed at:
Institut de Robòtica i Informàtica Industrial, CSIC-UPC

Thesis advisors:
Dr. Guillem Alenyà Ribas
Dr. Sergi Foix Salmerón

Dissertation Committee:
Prof. Dr. h.c. Michael Beetz (Universität Bremen)
Prof. Dr. Juan Antonio Rodríguez Aguilar (Artificial Intelligence Research Institute, CSIC)
Dr. Mohan Sridharan (University of Edinburgh)

..el científico busca lo común en lo diverso, separa lo esencial de lo superfluo; y es lo que continuamente hace Sancho Panza, que busca respuestas sensatas a los disparates de Don Quijote..

(entonces..la persona de ciencia, que formula disparatadas preguntas que luego responde, es a la vez Sancho, y también Quijote)

— Jorge Wagensberg Lubinski
(reflexión personal)

Abstract

A critical challenge in the design of robots that operate while interacting with humans is to ensure mutual understanding, which contributes to build reliable human-robot interactions. It is an arduous task since interactive scenarios are often uncertain, exposing robots to exogenous situations that affect their ongoing activities. In those cases, robots shall perceive and recognize unexpected changes in the environment, represent and reason about them, and decide how to adapt to them. This will certainly modify robots' internal knowledge, and it is fair to assume that part of the new robot beliefs might be hidden from other agents such as humans. Hence, robots shall also be capable of communicating or explaining the relevant knowledge about those beliefs updates. In this context, this thesis investigates the use of ontologies as an integrative framework for the construction of robot explanations, particularly within interactive settings involving humans. To this end, the thesis starts formulating the scope of the relevant domain knowledge to conceptualize, and it continues proposing novel ontological models and methods for ontology-based robot explanation generation. The first part of the thesis discusses two main contributions: a systematic review and classification of the state-of-the-art that narrows down the target set of reality phenomena to be conceptualized, and the investigation and development of novel robot perception methods to extract from realistic robot experiences the common patterns of the target conceptualization. The second part discusses the two remaining contributions: ontological analysis and modeling of the target domain knowledge, and the design and development of algorithms to construct ontology-based robot explanations. Note that the different ontological models and algorithms were mainly validated in collaborative and adaptive robotic scenarios. However, they were conceived from a foundational perspective, and we think that their scientific value extrapolates to other application domains (e.g. assistive robotics or non-robotic agents). Overall, the scientific contributions of this thesis set a solid foundational basis for the ontology-based explainable robots domain, boosting the design of trustworthy interactive robots.

Keywords: Applied ontology, Collaborative and Adaptive Robots, Explainable Agency, Ontology-based Explainable Robots.

Resumen

Un desafío crítico en el diseño de robots que interactúan con humanos es garantizar el entendimiento mutuo, lo que contribuye a construir interacciones fiables entre humanos y robots. Es una tarea ardua ya que los escenarios interactivos suelen presentar incertidumbre, lo que expone a los robots a situaciones exógenas que afectan sus actividades en curso. En esos casos, los robots deben percibir y reconocer cambios inesperados en el entorno, representarlos, razonar sobre ellos y decidir cómo adaptarse a ellos. Sin duda, esto modificará el conocimiento interno de los robots, y es justo suponer que parte de las nuevas creencias de los robots podrían estar ocultas a otros agentes como los humanos. Por tanto, los robots también deben ser capaces de comunicar o explicar el conocimiento relevante sobre las actualizaciones de esas creencias. En este contexto, esta tesis investiga el uso de ontologías como un marco integrador para la construcción de explicaciones de robots, particularmente dentro de entornos interactivos que involucran a humanos. Con este fin, la tesis comienza formulando el alcance del conocimiento del dominio relevante a conceptualizar y continúa proponiendo nuevos modelos y métodos ontológicos para la generación de explicaciones robóticas basadas en ontologías. La primera parte de la tesis analiza dos contribuciones principales: una revisión y clasificación sistemática del estado del arte que reduce el posible conjunto de fenómenos de la realidad a conceptualizar, y la investigación y el desarrollo de nuevos métodos de percepción de robots para extraer, a partir de experiencias realistas de un robot, los patrones comunes de la conceptualización objetivo. La segunda parte analiza las dos contribuciones restantes: el análisis ontológico y el modelado del conocimiento del dominio objetivo, y el diseño y desarrollo de algoritmos para construir explicaciones de robots basadas en ontologías. Cabe señalar que los diferentes modelos y algoritmos ontológicos se validaron principalmente en escenarios robóticos colaborativos y adaptativos. Sin embargo, fueron concebidos desde una perspectiva general y creemos que su valor científico se extrapola a otros dominios de aplicación (por ejemplo, la robótica asistencial o agentes no robóticos). En general, las contribuciones científicas de esta tesis establecen una base sólida para el dominio de los robots explicables basados en ontologías, impulsando el diseño de robots interactivos confiables.

Resum

Un desafiament crític en el disseny de robots que interactuen amb humans és garantir l'entesa mutu, fet que contribueix a construir interaccions fiables entre humans i robots. És una tasca àrdua ja que els escenaris interactius solen ser incerts, cosa que exposa els robots a situacions exògenes que afecten les seves activitats en curs. En aquests casos, els robots han de percebre i reconèixer canvis inesperats a l'entorn, representar-los, raonar sobre ells i decidir com adaptar-s'hi. Sens dubte això modificarà el coneixement intern dels robots, i és just suposar que part de les noves creences dels robots podrien estar ocultes a altres agents com els humans. Per tant, els robots també han de ser capaços de comunicar o d'explicar el coneixement rellevant sobre les actualitzacions d'aquestes creences. En aquest context, aquesta tesi investiga l'ús d'ontologies com un marc integrador per a la construcció d'explicacions de robots, particularment dins d'entorns interactius que involucren a humans. A aquest efecte, la tesi comença formulant l'abast del coneixement del domini rellevant a conceptualitzar i continua proposant nous models i mètodes ontològics per a la generació d'explicacions robòtiques basades en ontologies. La primera part de la tesi analitza dues contribucions principals: una revisió i una classificació sistemàtica de l'estat de l'art que redueix el possible conjunt de fenòmens de la realitat a conceptualitzar, i la investigació i el desenvolupament de nous mètodes de percepció de robots per extreure'n, a partir dexperiències realistes dun robot, els patrons comuns de la conceptualització objectiu. La segona part analitza les dues contribucions restants: l'anàlisi ontològica i el modelatge del coneixement del domini objectiu, i el disseny i el desenvolupament d'algorismes per construir explicacions de robots basades en ontologies. Cal assenyalar que els diferents models i algorismes ontològics es van validar principalment en escenaris robòtics col·laboratius i adaptatius. Tot i això, van ser concebuts des d'una perspectiva general i creiem que el seu valor científic s'extrapola a altres dominis d'aplicació (per exemple, la robòtica assistencial o agents no robòtics). En general, les contribucions científiques daquesta tesi estableixen una base sòlida per al domini dels robots explicables basats en ontologies, impulsant el disseny de robots interactius fiables.

Agradecimientos

En un lugar de la Mancha, de cuyo nombre siempre querré acordarme, no ha mucho tiempo que nació un niño de los comprometidos con la curisodiad, entregado a descubrir la verdad, y siempre lleno de sueños que lograr. Un lugar construido por gente de manos y rostros toscos, gente de una gran cultura natural, y una hospitalidad sin igual. Gente, que pareciera no hacer mucho ruído, y quizás por eso, gente que es capaz de escuchar. Un lugar manchego pintado en blanco y azul añil que se alza sobre una tierra roja, roja y no azul, como la sangre de aquellos que tuvieron la suerte de no nacer príncipes. Una tierra llana que se extiende hasta convertirse en horizonte, avivando así las utopías y los sueños de aquellos que se declaren intrépidos caminantes, como ya lo hizo el gran loco andante. Una tierra de gigantes, que se alzan desafiantes. Una tierra de vientos, doce, que mueven a su merced gigantes, para moler grano y alimentar los cuerpos de intrépidos caminantes. Fue en esa tierra, **Campo de Criptana**, La Mancha, donde nació y se forjó ese niño, es decir, yo, Alberto Olivares Alarcos. Por lo tanto, fue en esa tierra donde comezó a crearse esta tesis, porque no habría tesis sin el niño, porque no habría niño sin la tierra. Por eso, esta tesis empieza agradeciendo a la tierra que me vio nacer y a todas esas personas manchegas que me han acompañado en mi vida y mis sueños.

A mis padres, quienes siempre estuvieron de acuerdo en mostrarse generosos y solidarios con los demás, convirtiéndome en una persona enormemente alegre. «Por que la vida es darse. Darse, no hay alegría más alta», o eso me contó el compañero Eduardo Galeano. **Isabel Alarcos García** siempre fue un ejemplo de resiliencia y determinación, una luchadora feroz, una hormiguita eficaz y perfeccionista, una madre de cinco letras, entregada y cariñosa. **Luciano Olivares Díaz-Parreño** siempre fue un contador de historias formidable, y a su vez, un gran admirador del silencio y la escucha activa, un hombre de mirada y presencia poderosas, de sonrisa tímida, de seriedad irónica. **Isa** –así es como ella lo querría– **Olivares Alarcos** siempre fue un espíritu indomable, una llama que calienta y también quema, mi primera maestra, fuente de conocimiento del que no sale en los libros de texto, mi mejor amiga, mi confidente, mi protectora. **Diego Pintor Olivares** fue un recuerdo constante de equilibrio entre cuerpo y mente, un niño de frases concisas, un niño de tímida sonrisa, un niño que vino cuando más le necesitaba, a pesar de que pocos le esperaran, un niño que educó al hombre que soy, y por él hoy «quiero ser quien yo soy». **Manuel Antonio Rubio Palomino** fue mi primer mejor amigo, teníamos cinco años y era su cumple y los dos decidimos firmar el pacto que aún sígue vivo, él siempre fue un competidor honorable en mi afán de liderar y crear camino, siempre compartimos secretos e inquietudes, siempre me sentí protegido a su lado. **Francisco José Valero Díaz-Ropero** fue el primer amigo que sentí familia, hermano, antes de que llamarse así entre amigos fuera lo común, siempre nos recordaré cuidándonos el uno al

otro en la cotidianidad de nuestras tardes de tenis y noches de pelis. **Jesús Manjavacas Lucas** fue el primer amigo al que consideré «faro», un tipo con personalidad propia, que me enseñó a escuchar música, a digerir buen cine, siempre me sentí inspirado junto a él. **Enrique Cobos Manjavacas** fue amigo mucho antes de referirme a él como tal, compañero de campamentos por azar, durante años fue mi fiel escudero (o yo el suyo) en eso de aparentar menos edad de la que se pedía en las discotecas, con él siempre me sentí acompañado cuando quise debatir. **Alejandro Ramírez Martín** –¿o era Ramín Martínez?– fue el primer «lobo estepario» que descubrí, siempre dispuesto a sacarme de mis esquemas y también de cita por Madrid, a él le debo mi amor por la montaña, mi respeto a la montaña, mi pasión por la montaña, y es que siempre supo mostrarme lugares donde sentirme cobarde a su lado. **Pedro Sánchez Ramírez** fue el primer amigo que supo seguirme en lo que a locura se refiere, lo nuestro siempre fue perder calorías juntos, primero en la pista de baile, ahora corriendo; es cierto que lo abandoné en nuestra primera carrera juntos, pero siempre sabremos reirnos de ello. **Sergio Madrid Ucendo** fue el primero que me llamó «nene», provocando que casi una generación entera de mi pueblo me conociera como tal, él siempre me quiso como a un hermano pequeño. **David García Huertas** fue mi primer compañero de habitación –y el último–, con él compartí charlas filosóficas y fiestas sin igual, siempre me acordaré del tono de su voz y el acento de su habla. **Eduardo Parra Saavedra** fue el primer compañero de carrera y estuvimos juntos hasta el final; por un momento, llegué a envidiar su intelecto, aunque pronto me dediqué a admirarlo y a aprender de él tanto como pudiera. **Enrique López Hinarejos** fue el primer ser completamente puro que conocí; siempre me pareció increíble la de paz que desprende su sonrisa, el amor que transmiten sus ojos. Parra y Kike, Kike y Parra, me dieron aventuras manchadas de monedas al aire y locura en nuestros bailes. **Alberto Copado** –o simplemente Alber– fue mi primer compañero de festival, mi guía y protector en pogos, espejo y faro, hermano y maestro que siempre me ha cuidado. **Juan Álvarez Cabrerizo** –o mejor dicho, Fichaje–, fue mi primer compañero de piso, con él descubrí el gozo de cocinar y de los amaneceres metafísicos, siempre supimos entendernos a base de abrazos. Alber y Fichaje, siempre os amaré y respetaré porque si vuestra sonrisa mostrara el fondo de vuestro alma, la gente, al veros sonreír, lloraría con vosotros. **Sergio Santos** ha sido la única persona que aun sin tener doctorado ha contribuido de manera literal a esta tesis (Fig. 4.8); por ello y por hacerme tan feliz cada vez que nos vemos, le estaré siempre agradecido. **María Prieto Gutiérrez «la farandulera»**, fue la primera amante que vio a plena luz del día el lado oscuro de mi corazón, y ha sido la persona que ha visto (sin apartar la mirada) cómo me enfrentaba a gigantes y perdía. Ella es hogar, brasa incandescente que en silencio calienta, tierra roja que arraiga, agua clara que limpia. A menudo, pienso que mi naturaleza es existir y morir en guerra, terminar en el «Valhalla», pero ella me conduce a la paz que existe en el centro de mis tormentas. «No hay vida sin amor, al menos, no hay una vida que merezca la pena vivir» y yo siempre celebraré que tú seas el amor que alumbra la mía.

Todas estas personas, y quizás algunas otras que hoy olvido, han estado ahí cuando ni yo mismo quería; me rescataron muchas veces del encierro auto-exigido de las personas que se dedican a la ciencia. Ellas me enseñaron que el conocimiento antiguo quizás se encuentre en los libros, pero el conocimiento por descubrir sólo puede encontrarse fuera de ellos.

Hablando de conocimiento, creo que éste se descubre a solas, pero a menudo gracias a mentores. En mi caso, La Mancha me ofreció mentores fantásticos y ahora siento la necesidad de recordarles. **Mari Mínguez** fue la primera mentora, al menos que recuerde, que decidió darme

una oportunidad cuando yo no tenía claro si era merecedor de ella, me puso en contacto con el teatro, que se convirtió en la única fuente de felicidad que recuerdo de mi etapa escolar primaria. **Andrés Cintero** fue el primer mentor que me demostró que la risa y el aprendizaje pueden ir de la mano, para la memoria quedarán todas sus frases célebres que tanto me hicieron reír. **Luis Puebla**, que no Pueblas, fue el primer mentor que me enseñó la importancia de encontrar un equilibrio entre dureza y amabilidad; él supo enseñarme a pulirme a mí mismo. **Fernando Villanueva** fue el primer mentor que me trató como a un igual, aunque no lo fuéramos, él ha sido el mejor profesor de mi vida, supo motivar y guiar a un niño que andaba perdido. La vida se le escapó de entre las manos y hoy ya no hay alumnos que puedan disfrutar con él, pero le sentiré conmigo hoy. **Teo Casarrubios** fue el primer mentor que se convirtió en amigo, siempre admiré su sabiduría natural, aquella que sólo está al alcance de quien trabaja el campo, y siempre agradeceré sus consejos y apoyo. **Damián Ruíz** fue el primer mentor que sembró en mí la duda y el amor por la sabiduría, y por él comencé a razonar por el simple hecho de hacerlo, como si fuera un niño jugando, pero con la mente. **Andrés Salomón Vázquez** fue otro de esos mentores que confió en mí cuando ni yo mismo lo hacía, él me descubrió la robótica y me ofreció la oportunidad de investigar cuando yo más lo deseaba. **Paco Ramos** fue el primer mentor que me hizo sentir válido a pesar de mis debilidades, él me enseñó a ver mis fortalezas académicas y más tarde, fue quien me descubrió el mundo de las ontologías. Andrés y Paco son la representación viva de mi tesis doctoral, un matrimonio perfecto entre robótica y ontologías. Y aunque haya sido hace poco. **Manuel Ramírez Chicharro** fue el primer mentor con el que he sentido una conexión de hermandad; con él he hablado de ciencia, del futuro laboral, de filosofía, de historia, de nacimientos, de muertes, de vida, porque todo eso importa cuando se hace ciencia, porque la ciencia sólo puede entenderse en su contexto social.

Tal y como hizo el famoso hidalgo Alonso Quijano, encaminé mi andar hacia lejanas tierras. Fue así como llegué a Catalunya, con la esperanza, eso sí, de encontrar un sino distinto al suyo. Allí econtré un paisaje y cultura diferentes a los que me vieron nacer; la llanura se convirtió en montaña y el horizonte sólo podía verse cuando mirabas al mar. También econtré personas que me ayudaron a ser quien soy hoy, y por eso, es menester que les dedique unas palabras. **Salva Soler** fue el primer ser de luz que conocí, él me enseñó a actuar desde la verdad, me puso en contacto con mis emociones, me recordó que los humanos sabemos respirar con el diafragma de manera innata, e instauró en mí un mantra que hasta hoy me ayuda en mis momentos de duda «no me jodas, sigue y no te juzgues». **Héctor Uve** podría haber sido mi hermano gemelo si no fuera porque él es más fuerte y yo más guapo, entendemos los tempos el uno del otro, las respiraciones, las miradas, con él he compartido escenario, abrazos, llantos, risas y juegos de mesa. **Patrick Schneider** fue un gran descubrimiento para mí, amigo y apoyo en los momentos más duros y solitarios en Barcelona. Aprendimos juntos a hacer la compra y subsistir con poco dinero, y en él encontré esencias de otros amigos que estaban lejos. **Alejandro Suárez Hernández** fue el primer amigo que tuve en el IRI, de hecho, él fue quien me habló de la posibilidad de hacer la tesis aquí; con él entendí cuán doloroso puede ser vivir, aunque siempre supimos reirnos de ello. **Antonio Andriella** fue el primer compañero del IRI con el que tomé cervezas, después se convirtió en amigo, compañero de doctorado, y hasta de piso; mucho fue lo que aprendimos juntos, de ciencia, de vida, y lo que nos queda. **Alberto San Miguel Tello** –o como deberíais llamarle, Alberto Falso–, fue mi primer compañero de despacho y sin lugar a dudas ha sido mi gran compañero de tesis de principio a fin. De hecho, él terminó antes y tuve que imprimirme su cara y ponerla en un globo mientras escribía la tesis. Alberto, sé que

siempre jugamos a eso de que sólo puede quedar uno, pero ahora que estoy sin ti sé que la vida siempre es mejor contigo, te necesito (seguro que esta declaración me va a perseguir de por vida). **Luca Biccheri** è stato l'ultimo fratello di un'altra mamma che ho incontrato, con lui le notti di birre e filosofia non trovavano fine; con lui ho sempre sentito che non c'era Luca, né Alberto, ma qualcosa di più grande che ci nutriva tutti. Grazie per essere uno specchio, un faro e un fratello, un grande abbraccio, ragazzo fortunato. Por supuesto, he conocido a mucha otra gente fantástica en el IRI, **Sergi Martínez Sánchez** (amigo de charlas y compañero de viajes); **Pablo Salido Luis-Ravelo** (persona de enorme humor y amabilidad que hace que sea genial jugar en el lab); **Irene García Camacho** (persona de alta sensibilidad y caos); **Giorgis Tzelepis** (an enemy who reminds me of a friend); **Edoardo Caldarelli** (un matematico con il cuore di un musicista); **Oriol Barbany Mayor** (el déu del LaTeX); **Elena BBB** (alegre y poderosa como ella sola); **Aniol Civit Bertran** (un company que cuida la seva gent); **Tamlin Love** (a great actor and a better supreme leader); **Marc Gutiérrez Pérez** (un amor, aunque se empeñe en esconderlo). Finalmente, doy las gracias a la gente que facilita nuestra labor desde el laboratorio de Percepción y Manipulación, el taller y admin: **Sergi(s)**, **Patrick**, **Ferrán**, **Pablo** (otra vez), **Víctor**, **Diana**, **Edu**, **Mar**, **César**, **Carme**, **Andoni**, y **Patricia**.

En mis viajes, seguí encontrando mentores. **Horacio Rodríguez Hontoria** fue el primer profesor de Barcelona con el que me sentí en casa, él me mostró la importancia de buscar más allá de círculos de investigación endogámicos. **Guillem Alenyà Ribas** va ser el principal motiu pel qual vaig decidir fer el doctorat a l'IRI, de vegades, és fins i tot un reflex al mirall. És inspirador viure i treballar al seu costat, i li dec que ara sigui tan important per a mi «aprofitar el viatge». **Sergi Foix Salmerón** sempre juga en equip, encara que sempre va ser una persona difícil de convèncer, i gràcies a ell vaig aprendre a destil·lar el que escrivia i presentava, perquè la gent ho pogués entendre. Em va forjar com a investigador, però també com a persona. Gràcies, Guillem i Sergi, per acompanyar-me en aquest procés, per guiar-me quan estava perdut, per confiar en mi quan jo dubtava, per dubtar quan jo afirmava, i sobretot per cuidar-me. **Stefano Borgo** è diventato il mio modello di ricercatore, volevo essere come lui, provocatorio, rigoroso, umile, generoso, appassionato. Mi ha aperto la mente, ha cambiato il mio modo di vedere il linguaggio e di comprendere la realtà. È stata una responsabilità e un onore lavorare al suo fianco e essere trattato da pari a pari. Stefano, grazie per aver fatto parte dei momenti più felici e speciali del mio dottorato. **Gerard Canal Camprodon** va ser company a l'inici i després mentor, el meu doctorat va començar parlant amb ell d'ontologies i plans, i va acabar treballant amb ell integrant les dues coses, una manera bonica de tancar el cercle.

Of course, I am enormously grateful to the two reviewers of my thesis: **Michael Beetz** and **Mohan Sridharan**, their comments and suggestions have made me think of my research in a different way. Indeed, they both have influenced this thesis before sending me their comments. The very first time I read an article about ontologies for robots, or an article in general for that matter, was co-authored by Michael, and the first time I read an article about explainable agency was co-authored by Mohan. Isaac Newton said that scientists do research «on the shoulders of giants», thank you for being those giants for me.

A todos vosotros, kia ora!
A tots vosaltres, kia ora!
A tutti voi, kia ora!
To all of you, kia ora!

This work has been mainly supported by the European Social Fund and the Ministry of Business and Knowledge of Catalonia through the FI 2020 grant. Additionally, it has been also partially supported by:

- the Regional Catalan Agency ACCIÓ through the RIS3CAT2016 project SIMBIOTS (COMRDI16-1-0017);
- the Spanish State Research Agency through the María de Maeztu Seal of Excellence to IRI (Institut de Robòtica i Informàtica Industrial) (MDM-2016-0656) and the HuMoUR project TIN2017-90086-R (AEI/FEDER, UE);
- by MCIN/ AEI /10.13039/501100011033 under the project CHLOE-GRAPH (PID2020-119244GB-I00);
- by the European Commission NextGenerationEU, through CSIC's Thematic Platforms (PTI+ Neuro-Aging);
- by the European Union (EU) NextGenerationEU/PRTR and by MCIN/ AEI /10.13039/501100011033 under the project COHERENT (PCI2020-120718-2);
- and by the European Union under the project ARISE (HORIZON-CL4-2023-DIGITAL-EMERGING-01-101135959).

Acronyms

CORA Core Ontology for Robotics and Automation. 173, 175

DTW Dynamic Time Warping. 48, 51, 52, 53, 55

DUL DOLCE+DnS Ultralite Foundational Ontology. 172

FOL First-Order Logic. 17, 18, 19, 87, 91, 94, 98, 106, 107, 155, 168, 172

GPLVM Gaussian Process Latent Variable Model. 49, 50, 53, 54, 55, 56

kNN k-Nearest Neighbors. 48, 51, 52, 55

OWL Web Ontology Language. 17, 18, 98, 112, 172, 173

OWL 2 Second version of the Web Ontology Language. 113, 114, 140

OWL 2 DL Description Logics version of OWL 2. 87, 98, 99, 103, 106, 107, 114, 115, 133, 134, 137, 140, 155

OWL DL Description Logics version of OWL. 19, 168

PDDL Planning Domain Definition Language. 164

PPCA Probabilistic Principal Component Analysis. 49

RAG Retrieval Augmentation Generation. 158

RBF Radial Basis Function. 49

RDF Resource Description Framework. 18, 172, 174, 176

RDFS Resource Description Framework Schema. 137

ROS Robot Operating System. 165

SSM Speed and Separation Monitoring. 65

SSN Semantic Sensor Network. 165

SUMO Suggested Upper Merged Ontology. 19, 173, 175

SVM Support Vector Machine. 49, 55

TTC Time-to-contact. 10, 63, 64, 65, 66, 67, 68, 69, 73, 74, 75, 77, 79, 99, 119

UML Unified Modeling Language. 18

URDF Unified Robot Description Format. 165

Contents

Figures

Tables

chapter one

Introduction

> *..for it is owing to their wonder that people both now begin and at first began to philosophize..*
>
> **— Aristotle**
> (Metaphysics)

The integration of robotics into our daily lives is becoming increasingly apparent. It seems that this trend will continue to extend throughout our society, reaching a stage where robots engage with various entities during routine activities, such as collaborating with or aiding humans. Those interactive robots are expected to operate autonomously, which will surely require adapting the execution of their tasks, given the inherent high degree of uncertainty of interactive scenarios. Considering that robots will adapt to exogenous changes in their activity environment, it is fair to assume that they will update their knowledge so that part of it might be hidden from other agents such as humans. In order to ensure a reliable and proper interaction between the agents, mutual understanding is an essential prerequisite [Yuan et al., 2022]. This implies that robots shall understand the situations in which they are and also communicate and share such understanding with humans when needed. Hence, robots shall perceive and recognize unexpected situations in the environment, reason about how they affect their ongoing activity or task, and make reasonable decisions accordingly. Note that in those cases, the internal beliefs of robots will obviously evolve through perception and inference. Therefore, robots shall also be able to communicate or explain the details of those beliefs updates.

1.1 The role of applied ontology in human-robot shared understanding

Applied ontology builds on philosophy, cognitive science, linguistics and logic with the purpose of understanding, clarifying, making explicit and communicating people's assumptions about the nature and structure of the world [Oltramari, 2019]. A first definition of ontology was given by Gruber in [Gruber, 1993] stating: *an ontology is an explicit specification of a conceptualization*. Gruber's definition was informal and several authors tried to refine it. For instance, the notion of conceptualization was defined (mathematically speaking) as an intentional relational structure, i.e., a domain of discourse (a set of entities), a set of possible worlds (possible layouts of the entities), and a set of relations (stating which properties entities have in each possible world) [Guarino and Giaretta, 1995]. Others discussed further requirements for the definition like being *formal* and *shared* [Borst et al., 1997, Studer et al., 1998]. Finally, Guarino et al. [Guarino et al., 2009] settled the issue by proposing a formal definition which is today recognized in the community of applied ontology. An ontology is defined to be a *logical theory* consisting of a set of formulas whose models approximate as well as possible the intended models, i.e., those models that satisfy the conceptualization and the ontological commitments (the ontological principles one commits to).

Since an ontology is a logical theory, it can be used to represent knowledge and automate (deductive) reasoning for artificial agents and robots to infer conclusions from their current knowledge, for instance. However, ontological modeling is not just thought of as a tool for artificial agents such as robots, it would actually be useful for harmonizing people's assumptions about a target domain of discourse. Hence, when interactive agents agree on using the concepts defined in an ontology, they are in a position to share a mutual understanding of their experiences. In this thesis, logical formal languages are used to formalize the proposed models, which ensures a reliable use of the represented knowledge for sound robot reasoning. However, the use of a logic-based formalism might hinder the human comprehension of the robot's knowledge, thus, to ensure mutual understanding, the formalized knowledge is worked into explanations with a more natural format (e.g. textual natural language).

The process of conceptual ontological modeling (see Fig. 1.1) starts by focusing on a set of occurring reality phenomena to be conceptualized (e.g. the reflection of sky and mountains in lakes). Then, being exposed to those phenomena over time, it would be possible to extract patterns through perception. From the identified patterns, relevant 'invariants' of reality (things cognitively relevant for people) are isolated, comprising the elements or entities of the

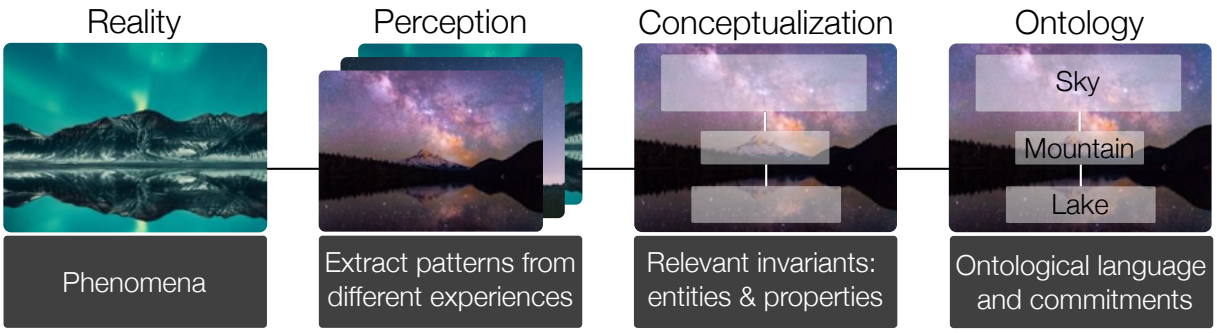


Figure 1.1: Ontological modeling process: from a target set of reality phenomena (left) to the ontology that models its conceptualization (right).

domain and the properties relating them. The set of those elements and properties is indeed the conceptualization, which is here understood as a language-independent abstraction and cannot be fully captured by any informal or formal language (e.g., natural, logical, etc.) [Oltramari, 2019]. Hence, the possible interpretations of the obtained conceptualization are constrained by the ontological commitments and using a specific language, obtaining the formulas that approximate as well as possible the intended models. Note that while ontologies are approximations of the conceptualizations they model, when carefully formalized, they are useful to harmonize agents' assumptions about a target domain of discourse. Indeed, this is not just a limitation of ontological models, humans can genuinely grasp a conceptualization that captures reality better than they can model and express it. In summary, *'all models are wrong, but some are useful'* (aphorism attributed to George E. P. Box).

Within the context of this thesis, the ontological modeling process starts by selecting a set of reality phenomena of interest from human-robot interaction scenarios. In order to make an informed selection, it may be useful to conduct a comprehensive study of the current state-of-the-art in the intersection of ontologies and robotics. Next, it is relevant to investigate robotic perception tasks in human-robot interactive experiences in order to identify patterns from the robots' perspective. Then, the sought conceptualization, a set of domain entities and properties relating them, may be obtained. Finally, by making some ontological commitments and selecting a language, the ontology can be formalized. Given the ontological model of the conceptualization, the data acquired through robot perception can be used to ground the obtained concepts, which is often known as the *symbol grounding problem* [Harnad, 1990]. Observe that symbols are only useful for a robot if they are grounded. A symbol such as *human collaborator* can only be utilized by a robot if there is a link between the symbol and the actual human in the real world. Using grounded knowledge, robots may understand the environment in which they are and make inferences and decisions, which will result in updating their

beliefs. Note that the process in which the perception data is abstracted may add some complexity to the robotic system, presenting some scalability issues. Indeed, abstracting all the data the robot is exposed to would certainly result in codifying huge volumes of seemingly irrelevant knowledge. This thesis aims to mitigate this by defining and grounding the ontological knowledge from actual robotic scenarios, while carefully selecting which knowledge is modeled. This way, the benefits of using an ontology outweigh those potential drawbacks. Furthermore, the ontological models are developed from a foundational viewpoint where the characterization of the core concepts is more important than the coverage of the application domain. Hence, the obtained models are small, general and still useful for realistic robotic applications, since they are properly scoped to a specific domain and carefully defined.

Note that human knowledge is often expressed symbolically, thus symbolic or logic-based representation and reasoning seem to make sense for human-robot shared understanding. However, there are well-known challenges and limitations inherent in the type of reasoning enabled by ontologies, and it is sensible to acknowledge them even though there is little or nothing one can do to solve them. For instance, ontology-based reasoning is precise because it provides us with certainty, but the real world is often full of uncertainties and ontologies cannot model what might be true. Hence, it is recommended to understand that ontologies are mostly useful to derive certain conclusions from a set of true statements, or even to identify incorrect conclusions or inconsistencies from false statements. However, ontology-based reasoning will never find general hypothesis or create new axioms based on experiential observations (i.e., inductive reasoning), nor will it help with seeking the most reasonable explanation to a specific event (i.e., abductive reasoning).

1.2 The role of applied ontology in explainable agency

In 2018, the European General Data Protection Regulation (GDPR) law [Carey, 2018] considered the right to explanations. Furthermore, the current success of 'black-box' machine learning models is increasingly making more evident the need for artificial intelligence systems to be explainable. Indeed, the European Union lawmakers reached a political agreement on the draft artificial intelligence (AI) act in December 2023. Proposed by the European Commission in April 2021, the draft AI act, the first binding worldwide horizontal regulation on AI, sets a common framework for the use and supply of AI systems in the EU [2021/0106(COD), 2024]. It offers a classification for AI systems with different requirements and obligations tailored on a 'risk-based approach'. In this context, research on eXplainable Artificial Intelligence (XAI) [Gunning and Aha, 2019] has recently drawn much attention, and the literature has

been populated with many works on easing the interpretation of artificial intelligent systems' decisions [Zhang and Zhu, 2018, Burkart and Huber, 2021]. In robotics, where robots not only make decisions but also act and produce changes in their environment, the need for explanations is even more justified. Indeed, the notion of explainable agency (i.e., explaining the reasoning of goal-driven agents and robots) has also gained significant momentum [Anjomshoae et al., 2019, Chakraborti et al., 2020].

There are three main elements of explainable agency: a *representation* of the content that supports explanations, an *episodic memory* to store agents experiences, and the ability to access the memory and retrieve and employ the stored content to *construct explanations* [Langley et al., 2017] (see Fig. 1.2). First, the *representation* shall include domain concepts and relations to describe the relevant knowledge generated during agents' experiences: the states perceived by agents (e.g. unexpected situations), the made decisions (e.g. adaptations), the criteria to make those decisions, etc. For this, both symbolic structures and numeric annotations will be required. Second, the *episodic memory* shall allow to record the states and relevant knowledge encountered during the experiences of agents. Episodic memory is the collection of past personal experiences that occurred at particular times and places [Tulving, 1972, Tulving, 2002]. In machine learning and other fields of artificial intelligence, there is no need for an episodic memory, because single post-hoc explanations of made decisions are often sufficient. However, in agency and robotics in particular, it is essential to connect the explanations to their specific context (e.g., place and time) within complete agents' experiences. Otherwise, the content of those explanations might not be properly understood. Third, explainable agents shall be able to *construct explanations* retrospectively, acquiring the content of their explanations from the episodic memory, and working it into a format that humans would find comprehensible. Furthermore, it shall be possible to select which and how much content is retrieved for the explanation generation, and the process may be interactive.

Readers might have already anticipated a plausible connection between these three elements and ontologies. Explainable agents shall be able to represent knowledge and data about their experiences, and an ontology is in essence a model of the knowledge of a domain of discourse (e.g. agents' experiences). Furthermore, since ontologies are usually defined using a formal logic language, it is possible to use the same language not only to store knowledge but also to retrieve it, which is related to the third element of explainable agency. Reflecting on these initial observations, one might wonder whether applied ontology might become the unifying component of the three elements of explainable agents. In this regard, some authors have recently investigated the strong relation between semantics, ontology and explanation especially under particular interpretations [Guizzardi and Guarino, 2023]. There are also some attempts to formalize the notion of explanation ontologically [Tiddi et al., 2015], which can be used to help

Figure 1.2: Elements of explainable agents: a representation of the domain knowledge and data, an episodic memory to store experiential knowledge, and the ability to select knowledge from the memory and to construct the explanation.

system designers to make decisions about the explanations to include in their systems [Chari et al., 2020]. Furthermore, other knowledge-based representations (e.g. non-monotonic logic) have been lately used to support explainable agency [Sridharan, 2023]. However, to the best of our knowledge, the literature does not contain works that deeply investigate the role of ontologies in the three elements of explainable agency as a whole. Indeed, as it is discussed in Chapter 2, ontologies have neither been used for explainable robotics. For this reason, the research presented in this thesis focuses on the union of applied ontologies and explainable robotics, aiming to foster a shared understanding in human-robot interactive experiences.

1.3 The role of explainable agency in framing the scope of the ontological model

Explainable agents require a representation and an episodic storage of knowledge about their experiences, which will be later queried to construct explanations. Hence, if one wants to use ontologies to represent such knowledge, the ontologies' scope will certainly be conditioned to capture the desired explanation content. Apart from the three main elements of explainable agency, Langley [Langley et al., 2017] also defended the idea that explainable agency requires four distinct functional abilities. Based on that work, explainable agents shall: report the actions they executed, explain how actual events diverged from what was planned and how they adapted to it, explain decisions made during plan generation (comparing alternatives), and communicate all of this in a way that is close to human concepts.

In order to exhibit those functional abilities, ontology-based explainable agents shall count on an ontological model that can support them. Hence, the ontology should capture concepts and relationships modeling the main knowledge around their executed actions, the notions regarding unexpected situations and the undertaken adaptations, and how different alternative plans compare. Furthermore, explainable agents should be able to work such ontological knowledge into a format that ensures human understanding of the agents' beliefs. These ideas were considered when deciding the scope of the ontological models proposed in this thesis. Indeed, the thesis includes works that address the four functional abilities of explainable agents.

1.4 Contributions

This work examines the use of ontologies as an integrative framework for the construction of robot explanations, particularly within interactive settings involving humans. In the following, a summary of the contributions of this work is provided.

1.4.1 Main scientific contributions

C1 Systematic review and classification of the state-of-the-art works that use ontologies in robotics to support robot autonomy [Olivares-Alarcos et al., 2019a]. This contribution helped narrow down the target set of reality phenomena to be conceptualized, focusing primarily on those encountered within collaborative robotic scenarios.

C2 Investigation and development of novel robot perception methods for recognition and decision-making tasks in collaborative robotics scenarios, specially in cases where humans and robots closely interact. The approaches focus on human intention and risk of collision recognition, extracting the common patterns of those robotic experiences that will be later conceptualized [Olivares-Alarcos et al., 2019c, Olivares-Alarcos et al., 2023b].

C3 Ontological analysis and conceptual modeling of the inherent knowledge found during the execution of collaborative and adaptive robot experiences [Olivares-Alarcos et al., 2022], and the relevant knowledge related to robot selection from alternative plans [Olivares-Alarcos et al., 2024]. These models harmonize the terminology and provide robots with knowledge representation and reasoning tools in their respective domains. Hence, robots may now reason about whether some events are or not collaborations, which events are adaptations, or how different plans compare to each

other (e.g. one is better than the others). Note that this knowledge will indeed be useful to support all the functional abilities of explainable agents proposed in the literature.

C4 Design and development of ontology-based algorithms for the construction of robot explanations of collaborative and adaptive experiences [Olivares-Alarcos et al., 2023a], and contrastive explanations of competing robot plans [Olivares-Alarcos et al., 2024]. Both algorithms are general enough to work with any of the ontological models proposed in the thesis, and even other models as long as they are formalized in the same ontological language. The algorithms leverage the structure of ontological knowledge to build different types of explanations for different purposes: plain narratives and contrastive explanations.

1.4.2 Main technical contributions

Together with some of the main scientific contributions, this thesis also produced a set of open access technical contributions.

- Knowledge-based framework for collaborative robotics and adaptation (*know-cra*).[1]
- Knowledge-based framework for the generation and comparison of robot plans (*know-plan*).[2]
- Ontology-based explainable robots framework for collaborative and adaptive experiences (*XONCRA*).[3]

1.4.3 Collaborative scientific contributions

The investigation conducted during this thesis has also influenced and contributed to some works that were done in collaboration with other researchers.

- Investigation and development of alternative robot perception methods to the ones presented in the main contributions [Maceira et al., 2020, Gassó Loncan Vallecillo et al., 2020].
- Development of the IEEE 1872.2-2021 Standard for Autonomous Robotics (AuR) Ontology [IEEE-SA, 2021, Gonçalves et al., 2021a]. The standard specified ontological concepts for AuR influenced by the findings of the review presented in the main contributions [Olivares-Alarcos et al., 2019a].

[1] https://github.com/albertoOA/know_cra
[2] https://github.com/albertoOA/know_plan
[3] https://github.com/albertoOA/explanatory_narratives_cra

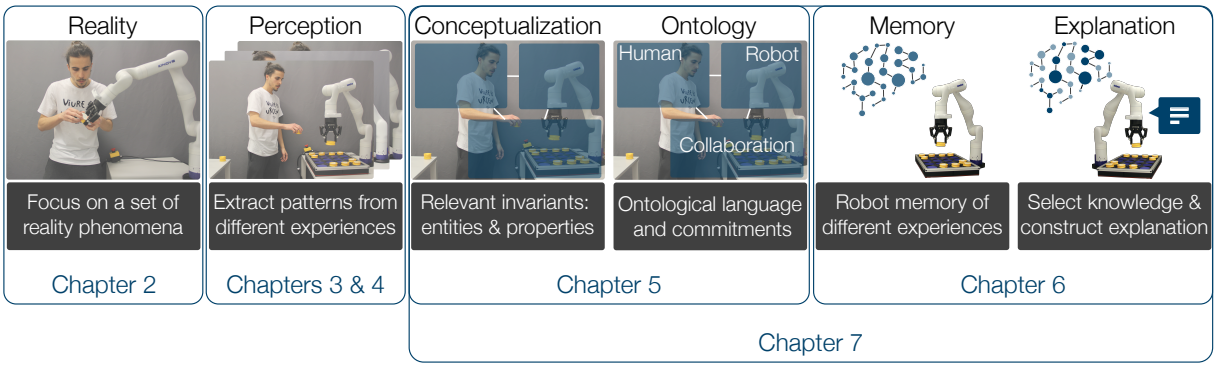


Figure 1.3: Ontology-based explainable robots: from a selected set of reality phenomena to their corresponding explanation, through the ontology that models their conceptualization and supports the explanation's construction.

1.5 Outline

The remainder of this thesis is organized in two parts each consisting of different chapters that all contribute to the overall goal of investigating and establishing the foundations of ontology-based explainable robots. In the following, a short abstract will be provided for each of the remaining chapters that contribute to this field. Fig. 1.3 combines the content depicted in Figs. 1.1 (ontological modeling process) and 1.2 (explainable agency elements), providing a visual overview of the elements of ontology-based explainable robots and how they relate to the structure of this thesis.

Part I explores and fixes the overall scope of the target robotic phenomena and ontological domain to conceptualize. First, through a comprehensive review of the state-of-the-art, and then with the investigation of robot perception methods for a reliable recognition and categorization of interactive experiences with humans.

- **Chapter** 2 explores the literature on ontology-based approaches to robot autonomy, comprehensively reviewing a set of existing frameworks. This allows to make an informed decision to constrain the scope of the intended ontological conceptualization, focusing on industrial collaborative robotics phenomena.

- **Chapter** 3 investigates robot perception approaches to recognize and classify human intentions in a collaborative scenario. It introduces a novel dataset of interactive force-based information for human-robot collaboration, and evaluates two different force-based human intention recognition methods. Finally, the approach's ability to generalize to different users is studied (N=15). The chapter finishes with a short discussion on important insights into common patterns to be conceptualized, which were

extracted from the perceived phenomenon.

- **Chapter** 4 focuses on robot recognition and classification of different levels of human-robot collision risk. It reveals how to extend a two-dimensional formulation to compute time-to-contact (TTC) in a three-dimensional space, and how to stop the robot's motion based on the computed TTC. The proposal is evaluated both in simulation and on a physical robot. The chapter ends by briefly arguing on significant intuitions about patterns to be conceptualized, which emerged from the hands-on experience. Indeed, the robotic task introduced in this chapter is later used during the conceptual modeling and ontology-based explanation generation discussed in Chapters 5 and 6 respectively.

Part II is devoted to the ontological conceptual modeling, and the investigation of the use of the obtained models for explainable agency in robotics.

- **Chapter** 5 conceptualizes and formalizes an ontological model for collaborative robotics and adaptation, OCRA, the very first ontology for reasoning about both human-robot collaboration and robot plan adaptation. The chapter shows how OCRA can be used to recognize and categorize collaborative and adaptive events, being able to answer a set of competency questions. For this, it is used the prototypical collaborative task from Chapter 4, thus the proposed symbolic model is directly grounded in the task's data. Finally, the ontology robustness is evaluated in some limit cases of the formalization.

- **Chapter** 6 addresses the construction of robot explanatory narratives of collaborative and adaptive experiences within the same scenario used in Chapters 4 and 5. Hence, the explanations are built with both abstract knowledge and data from a realistic collaborative task. The chapter explains the integration of the OCRA ontology in an episodic memory framework, and proposes a novel algorithm (AXON) for the narrative generation. The perceived narratives' usefulness is assessed through a pilot study with users (N=30). This work represents the first attempt to propose a framework for ontology-based explainable robots.

- **Chapter** 7 extends the work presented in Chapters 5 and 6 from collaborative and adaptive experiences to cases in which robots compare and explain the differences of competing plans. It introduces a new ontology to model the properties of plans and to reason about how different plans relate to each other, and a novel algorithm for contrastive ontology-based explanations (ACXON). The chapter tackles the functional abilities and explanation features that were missing in the previous works of the thesis. Hence, culminating the main objective of the thesis: setting the foundational basis of ontology-based explainable robots.

- **Chapter** 8 concludes the thesis, discussing the most important findings of this research, and identifying open challenges, providing directions for future work.

Part I

Formulating the scope of the ontological conceptualization

chapter two

Reviewing ontological models for autonomous robots

> *..creative thinking enters far more into problem formulation that it does into problem solving, problem formulation is often by far the trickier part..*
>
> **— Murray Gell-Mann**
> (Peace Summit 2009 - Educating the Heart and Mind)

This thesis aims to investigate and establish the foundations of ontology-based explainable robots. The introduction discussed in Chapter 1 suggests starting from an ontological modeling process, specifically, by focusing on a set of occurring reality phenomena to be conceptualized. In the thesis context, it would make sense to choose such reality phenomena from human-robot interactive scenarios, where explanations would enhance the interaction. In order to make an informed selection, it may be beneficial to conduct a comprehensive study of the current state-of-the-art in the intersection of ontologies and robotics. This would be useful for identifying how ontologies are employed in the domain and pinpointing sub-domains that might be worth investigating in detail. Hence, this chapter reviews the literature concerning the development and use of ontologies for robotic applications. Particularly, it introduces a classification of ontology-based approaches to robot autonomy, and discusses and compares those approaches. Note that the focus is on frameworks that employ ontologies to support robot autonomy rather than on particular ontologies that are designed for robots. At the end of the chapter, the findings of this research are comprehensively analyzed and discussed, paying

special attention to how they relate to the main topics addressed during this thesis. For instance, the review revealed the lack of works conducting research on ontologies for collaborative and industrial robotics, which was considered when selecting the target reality phenomena to be conceptualized. Furthermore, no works were found that investigate the role of ontologies in enabling robots to construct explanations, highlighting the necessity of establishing the foundational aspects of ontology-based explainable robots.

2.1 Motive

Before investigating the role of ontologies in building explainable robots, it is vital to acquire a thorough understanding of how ontologies have been used in the robotics literature, identifying good practices and things to improve or to be done. The literature already contains some efforts to survey the current research on the union of ontologies and robotics, which may provide part of the desired understanding. Some of those surveys focused on particular robotic tasks, and compared different knowledge-based approaches with respect to their suitability for the investigated task. For example, Thosar et al. [Thosar et al., 2018] investigated the suitability of nine robot knowledge bases in a household scenario, and for the task of replacing missing objects that play a role in a task (e.g. a tool) with similar ones. They analyzed the amount of knowledge represented in each of the frameworks and their research impact through their number of citations. Each knowledge base was further studied with respect to the following criteria: knowledge acquisition, representation formalism, symbol grounding, modeling of uncertainty, and the inference mechanism. Another interesting study [Paulius and Sun, 2019] provided an analysis of different knowledge representation aspects related to service robotics. The authors first gave an overview of knowledge representations, with special focus on cloud-based knowledge representations and cognitive architectures. Then, they examined several knowledge-based models and their role in activity understanding and task execution. Finally, it was argued that machine learning methods can complement knowledge representation approaches, and they presented a set of key components for effective knowledge representation for robots.

The two surveys provided a good amount of information about ontologies for robotics, but focusing on narrow scopes: for specific tasks (e.g., object substitution or activity understanding), and particular application domains (e.g. household or service robots). Indeed, none of them tackled aspects that are relevant to this work such as: how ontologies can be used, for instance, to create robots' memories, or to generate explanations. Those and other questions are still open, thus, this chapter conducts a review from a more general viewpoint, drawing a landscape

of how ontologies are used to support robot autonomy.

2.2 Basics of ontology and autonomous robotics

In this section, we first introduce ontology as a knowledge artifact, and list some ontology classifications that are relevant for later discussion. Then, we review the terminology used in autonomous robotics, and some of the (computational) problems and capabilities that are considered essential for agents' autonomy.

2.2.1 Ontologies

Recall that an ontology is defined to be a *logical theory* consisting of a set of formulas whose models approximate as well as possible the intended models, i.e., those models that satisfy the conceptualization and the ontological commitments (the ontological principles one commits to) [Guarino et al., 2009]. Being a logical theory, it consists of individuals, classes, functions, relations and axioms. The exact list changes depending on the specific logic language one adopts. Usually, an ontology is given in First-Order Logic (FOL), or in Web Ontology Language (OWL). *Individuals* are the objects in the ontology, the things the ontology is about. *Classes* are properties and are used to identify the individuals that satisfy that property. *Functions* are formed from certain relations and can be used in place of an individual. *Relations* are connections across individuals. *Axioms* are expressions in the language that use the previous elements to state what is true in the ontology.

In domain studies, the term ontology is used to refer to a variety of things. For instance, for Chandrasekaran et al. [Chandrasekaran et al., 1999] an ontology is a *representation vocabulary* specialized to some domain and constrained by a conceptualization. It is also understood as a *domain theory* about objects, properties and relationships among those objects that are possible within a specified domain of knowledge. The purpose of an ontology in this sense is to provide the knowledge structure for a particular domain, therefore, it focuses only on the viewpoint taken within the domain, and it includes the relevant concepts for working in such domain. In the literature, this latter use of the term is known as *domain ontology* while the characterization introduced by Guarino and colleagues [Guarino et al., 2009] is general encompassing domain as well as foundational ontologies that contain very general terms applicable across all domains.

Developing a domain ontology, one provides the description of a particular domain without challenging the ontological perspective. The purpose is to make the domain knowledge explicit and formal, i.e., to fix in a formal language the vocabulary and what the experts consider its correct interpretation including the valid assertions in the domain. A domain ontology can help

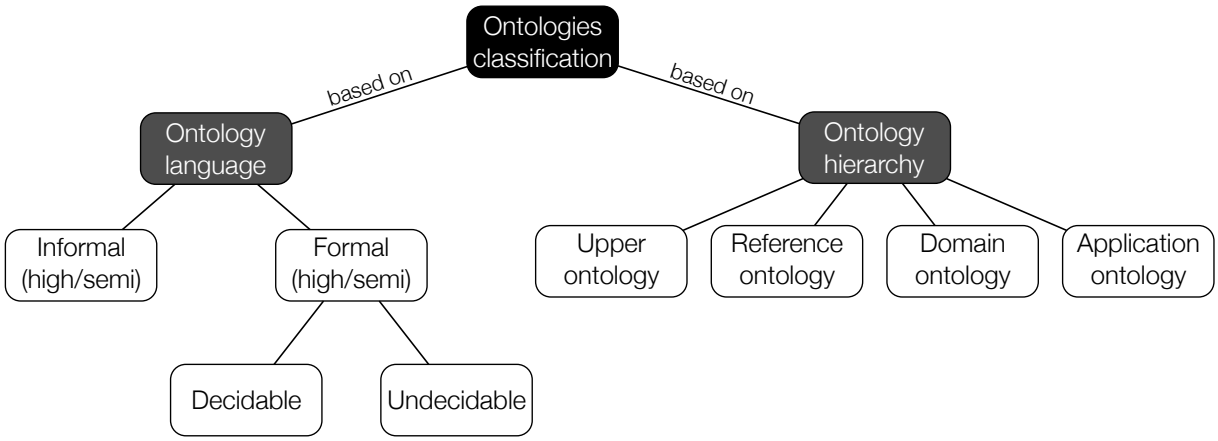


Figure 2.1: Different classifications of ontologies explained in this work. They are based on: the language used to write the ontology, and the hierarchical level of the ontology.

to achieve data and model interchangeability within and across communities.

It is worth noting that there is no unique conceptualization of a domain. Having suitable domain ontologies helps to clarify the differences as well as to compare the conceptualizations.

Types of ontologies

Ontologies can be classified along many dimensions, in this thesis we will consider the ones depicted in Figure 2.1.

A classification based on the characteristics of the language used for the ontology is presented by Uschold and Gruninger [Uschold and Gruninger, 1996]. It shows that the term 'ontology' is sometimes used vaguely. That classification divides ontologies into four classes: highly informal, semi-informal, semi-formal, and rigorously formal. However, following our previous discussion of what is an ontology, we observe that the first class is not talking about ontologies: it refers to linguistic resources or to knowledge repositories in an early phase of ontology construction.

Since the language of the ontology constrains how the ontology can be used in an information system, this kind of classification is of primary relevance in the context of this chapter. The first distinction we introduce is between informal and formal languages. By *informal* we mean a language that does not have an associated formal semantics, like Resource Description Framework (RDF), (part of) Unified Modeling Language (UML). They are mostly dedicated to representation tasks and syntactic manipulation. Automatic reasoning in these languages is not reliable because there is no systematic way to constrain their interpretation. By *formal* we mean a language endowed with formal (e.g., Tarskian) semantics, that is, languages whose interpretation is formally established. These languages, among which we find FOL and OWL, are suitable for knowledge representation and reasoning since they are based

on clear and exhaustive syntactic and semantic rules. They are among the most reliable languages we have available in knowledge modeling. Among the formal languages, a further distinction is of interest to us: decidable vs undecidable languages. Here decidable means that, given a logical theory expressed in that language, there exists a method for determining whether an arbitrary formula is derivable or not in the theory. Since ontologies are special logical theories, an ontology written in a decidable language is decidable. A language is called undecidable if it is not decidable. An ontology that uses a decidable language is also called *computational* since it can be used at run-time for information extraction and verification, e.g., OWL DL. A formal ontology, which is not computational, like FOL, is not appropriate for such a role since, when queried, it might not return an answer. Since for application purposes, the answer to a query might need to be available very quickly, decidability is enriched with computational complexity considerations [Papadimitriou, 2003].

Based on hierarchy, ontology classifications usually divide ontological systems into upper-level, reference, domain and application. An *upper-level ontology* is an ontology that focuses on widely applicable concepts like object, event, state, quality, and high-level relations like part-hood, constitution, participation, and dependence. Examples are SUMO (Suggested Upper Merged Ontology) [Niles and Pease, 2001], Cyc ontology [Elkan and Greiner, 1993], BFO (Basic Formal Ontology) [Arp et al., 2015] and DOLCE (Descriptive Ontology for Linguistic and Cognitive Engineering) [Borgo et al., 2021]. A *reference ontology* is an ontology that focuses on a discipline with the goal of fixing the general terms in it. It is highly reusable within the discipline, e.g., medical, engineering, enterprise, etc. [Guarino, 1998]. When the ontology focuses on a more limited area, e.g. manufacturing or robotics, we call it a *domain ontology*. This kind of ontology provides vocabulary about concepts within a domain and their relationships, about the activities taking place in that domain, and about the theories and elementary principles governing that domain. The concepts in domain ontologies are mostly specializations of concepts already defined in upper-level and reference ontologies, and the same might occur with the relations [Gómez-Pérez et al., 2004]. An *application ontology* contains all the definitions needed to model the knowledge required for a particular application, e.g., a polishing or kitting robotic system. In this regard, this thesis contributes to novel domain and application ontologies.

2.2.2 Autonomous robotics

Autonomy is a desirable quality for robotic agents in many application domains, especially when the robot needs to act in real-world environments together with other agents, and when the environment changes in unforeseeable ways. Robot autonomy is further critical when

employed under certain legal and ethical constraints (e.g., a robot assistant at the hospital, or an industrial collaborative robot cooperating with humans). Apart from the dictionary and subjective definitions, there exist several attempts to define the term. For instance, the ISO 8373:2021 [ISO 8373:2021, 2021], which defined robotics vocabulary, included a definition of autonomy. However, no broad consensus on this matter has been reached so far. Indeed, Beer et al. [Beer et al., 2014] presented a comprehensive analysis of existing definitions in several domains including robotics, and proposed their own definition of *autonomy*:

The extent to which a robot can sense its environment, plan based on that environment, and act upon that environment with the intent of reaching some task-specific goal (either given to or created by the robot) without external control

Computational problems and capabilities

Autonomous systems can be based on different architectures with different levels of complexity ranging from simple reactive architectures to deliberative architectures or cognitive architectures. Reactive systems are based on simple *sense-act* loops, while deliberative systems employ more sophisticated *sense-decide-act* loops to endow the system with reasoning and decision-making capabilities. However, it is still an open question what the computational capabilities are that enable human-level cognition, and how these are structured in a cognitive architecture. Vernon et al. provided a thorough discussion about this topic [Vernon et al., 2007]. The authors presented a broad survey of the various paradigms of cognition, addressing cognitivist approaches (physical symbol systems), emergent, connectionist, dynamical, and enactive systems, and also efforts to combine the different approaches in hybrid systems. Then, a review of several cognitive architectures drawn from these paradigms was surveyed. An extension of that survey was provided by Vernon [Vernon, 2014]. In that last work, Vernon referred to the key architectural features that systems capable of autonomous development of mental capabilities should exhibit [Langley et al., 2009].

In this work, we follow Langley's thoughts, precisely concerning about *what* are the functional capabilities an autonomous robot should demonstrate, more than *how* those functional modules should interconnect [Langley et al., 2009]. Therefore, instead of proposing a novel cognitive architecture, which is out of the scope of this chapter, we only make sure to address all terms related to those functional capabilities provided in Langleys's work, which are explained in the Section 2.3.2 and listed below:

1. Recognition and categorization;
2. Decision making and choice;

3. Perception and situation assessment;
4. Prediction and monitoring;
5. Problem solving and planning;
6. Reasoning and belief maintenance;
7. Execution and action;
8. Interaction and communication; and
9. Remembering, reflection, and learning.

2.3 A classification of ontologies for autonomous robots

In this section, we present the classification that will be utilized to structure and perform the review of the selected works in Section 2.4. The classification is split into three dimensions: (a) ontology scope, (b) reasoning scope and (c) application domain scope.

2.3.1 Ontology scope

Ontologies can be organized as networks of modules, which can themselves be ontologies, each focusing on a specific topic. The scope of a module is given by the range of categories that it covers. In this section, we propose a list of categories particularly relevant in autonomous robotics such as `Sensor`, `Capability`, and `Action`. Our aim is to find how these categories have been used in the literature and to provide an initial discussion about their actual meaning. References to more detailed discussions in the literature are also provided. The Oxford dictionary [OED, 2024] is used to provide informal definitions, to discuss how the terms are understood in common and conventional ontologies, and to highlight the different usage in the robotics field. In Section 2.4.1, we analyze whether or not each of the surveyed projects defines these categories and how.

Note that many concepts relevant to autonomous robotics lack a universally agreed meaning. Moreover, terms commonly used in the robotics field are also often part of everybody's everyday speech. Hence, everybody has *some* intuitive definition of these terms that is often heavily influenced by personal experience, and thus it may be substantially different from that of other people. Appendix B.1.1 introduces the complete list of considered categories with their most widely agreed meanings, and provides a brief comparison between the different viewpoints.

2.3.2 Reasoning scope

This scope of reasoning is our second classification criterion for the comparison between ontology-based approaches in autonomous robotics. It is considered a categorization of ontology-based reasoning tasks that are in particular relevant for autonomous robotics, which have been considered in previous works. Indeed, this categorization is based on the nine functional capabilities presented in Section 2.2.2 which every autonomous robot should exhibit [Langley et al., 2009]. In Section 2.4.2, we discuss how the surveyed projects use ontologies to support each of these nine capabilities.

One of the keys to the success of knowledge-based approaches in autonomous robotics, is the use of ontologies as *enabler* to help the robot to understand and reason about its environment when executing tasks. For instance, implementing robotic applications that would not be possible without the use of KR techniques, and which substantially enhance the state of the art in autonomous robotics. The main research question is how the different software components of integrated robot control and perception systems could be enhanced through the use of ontologies and automated reasoning. In Appendix B.1.2, readers can find a definition of the nine capabilities and an intuition of how ontologies can support them.

2.3.3 Application domain scope

The last classification criterion for the comparison between ontology-based approaches in autonomous robotics is regarding to the application domain. In this section, we comment some of the different application domains of robotics and the two principal domains we consider. In Section 2.4.3, we discuss for which of the two domains each of the reviewed projects has been designed and used.

Robotics is a multidisciplinary and versatile discipline whose application is present in wide range of domains: Medicine, Industry, Assistance, Entertainment, Space, Military. Therefore, it exists such a broad spectrum of application domains for robotics, that it is not possible to go through all of them without excessively extending this chapter. In accordance with this thought and following the classification of robotics devices published in the ISO 8373:2021 [ISO 8373:2021, 2021], we focus on two domains: Industrial and Service robotics. They both include, if not all at least many, of the application sub-domains of interest for robotics (e.g. medicine, military, collaboration, assistance, rescue, social, etc.). The ISO 8337:2021 specifies a vocabulary used in relation with robots and robotic devices operating in both industrial and non-industrial environments (service). It also provides definitions and explanations of the most commonly used terms. In Section B.1.3 readers can find the definitions for Industrial and Service robot, which were partially extracted from the ISO.

2.4 Ontologies to support robot autonomy: literature frameworks comparison

In this section, we provide a discussion and comparison of frameworks that use ontologies to support robot autonomy. We perform a literature search restricted to a set of criteria such as ontology scope, curation, or accessibility, see Appendix B.2.1 for the complete description. For each framework that fulfills these criteria, it is provided a brief discussion in Appendix B.2.2. It is also provided a short justification of why some relevant projects that do not fulfill all criteria were not included (see Appendix B.2.3).

For the purpose of comparing the frameworks and projects described in Appendix B.2.2, in this section we explore how each of the projects addresses the different aspects included in the classification of ontologies proposed along the Section 2.3. Specifically, the ontological, reasoning and application domain scopes of each of the selected works are thoroughly examined and contrasted.

2.4.1 Comparing the frameworks based on their ontology scope

In this section, it is explored which of the terms discussed in the Section 2.3.1 are defined in each of the selected frameworks/projects. Note that we only consider that a term is defined when the natural language definition and/or the formalization are compliant with our domain. Indeed, if the exact term is not defined but the desired notion is captured by a similar term, for us, the concept is defined. Table 2.1 provides an overview of the concepts defined in each of the works.

Object Most of the compared frameworks, define `Object` from an endurant perspective. In both, KnowRob 1.0 and ORO, the exact term is not defined but it is used (from Cyc ontology) the notion of `Spatial Thing`: *The collection of all things that have a spatial extent or location relative to some other Spatial Thing or in some embedding space*. OROSU uses SUMO's definition: *Corresponds roughly to the class of ordinary objects. Examples include normal physical objects, geographical regions, and locations of Processes, the complement of Objects in the Physical class*. ROSETTA does not concern about spatial regions and only focuses on `Physical Objects`: *Every automated work cell consists of physical objects. Some objects, devices, are active and have skills, while other, such as work pieces, are passive and are manipulated by the devices*. PMK proposes the use of `WSObjectClass`, which is split into: `Artifact`, `Artifact Components` and `Collections`. For example, a cup (artifact) is an object that has body and handle (artifact components) and could be served with saucer (collection). On the other hand, KnowRob 2.0,

Term	KnowRob 1/2	ROSETTA	ORO	CARESSES	OROSU	PMK
Objects	**Yes**/**Yes**	**Yes**	**Yes**	No	**Yes**	**Yes**
Environment map	**Yes**/**Yes**	No	No	No	**Yes**	**Yes**
Affordance	No/**Yes**	No	**Yes**	No	**Yes**	No
Action	**Yes**/**Yes**	No	**Yes**	**Yes**	**Yes**	**Yes**
Task	No/**Yes**	**Yes**	**Yes**	No	No	**Yes**
Activity	No/No	No	No	**Yes**	No	No
Behavior	No/No	No	No	No	No	No
Function	No/No	No	No	No	No	**Yes**
Plan	No/**Yes**	No	**Yes**	No	No	No
Method	No/**Yes**	No	No	No	No	No
Capability	**Yes**/**Yes**	**Yes**	No	No	No	**Yes**
Skill	No/No	**Yes**	No	No	No	No
Hardware	**Yes**/**Yes**	**Yes**	**Yes**	No	**Yes**	**Yes**
Software	**Yes**/**Yes**	**Yes**	No	No	**Yes**	**Yes**
Interaction	No/No	No	No	No	No	No
Communication	**Yes**/No	No	No	No	No	No

Table 2.1: List of relevant terms for the autonomous robotics domain, and their coverage in the different chosen works. *Yes* and *No* state for when the term is or not covered by the ontology of the specific framework. Note that in the cases when the term is needed and taken from the upper ontology used within the framework, and/or when the knowledge is captured using a similar term, it is considered that the term is covered. If the upper ontology contains the term but it is not used, we consider that the term is not included.

based on the DUL Ontology, considers not only physical entities: *Any physical, social, or mental object, or a substance. Objects are always participating in some event (at least their own life), and are spatially located*. In CARESSES, we cannot find any definition in natural language but object is defined as a subclass of the entity `Topic`, which is *any theme a robot can talk about*. We understand that this last definition is not aligned to what is needed in our domain. Hence, we claim that CARESSES does not define `Object`.

Environment map In both versions of KnowRob, it is possible to find the concept of `SemanticEnvironmentMap` as a sub-class of `Map`. However, there is not any natural language definition and it is not aligned to DUL (current upper ontology) yet. OROSU defines places and environments where the robot works (e.g. `CTRoom`, `EngineeringRoom`, `OperatingRoom`, which are sub-classes of `Room` and are connected to actions which are expected to take place in there. PMK formalizes the notion of `Workspace` which has three gradual sub-classes, `Region` (i.e., free and occupied regions), `Physical Environment` (topology of the environment entities), and `Semantic Environment` (semantic information of the workspace).

Affordance The concept of `Affordance` is not exactly defined in any of the works which are object of study, still, it is possible to find some definitions that partially capture the same knowledge. ORO defines the property `canBeManipulated` which *indicates if an object can be manipulated and that the agent knows a grasping point for the object. Thus, if the object can be manipulated, it is movable as well*. OROSU also describes a property, in this case `CanGrab`, which *indicates that a device can grab an object*. Even though in the latter it is not mentioned the existence of any grasping point, it is implicit that if a robot can grasp an object it also knows where to do it from. KnowRob 2.0 has recently also introduced a notion of affordance that can be found in published ontologies, and also in a scientific publication that discusses this concept [Beßler et al., 2020a]. In KnowRob 2.0, affordances are defined as *the description of a property of an object that can enable an agent to perform a certain task*.

Action KnowRob 1.0 uses the definition provided by the Cyc ontology in which `Action` is defined as *an `Event`*. ORO, while also using the formalization from Cyc, provides a more concrete natural language definition of the term: *The collection of Events that are carried out by some "doer". Instances of Action include any event in which one or more agents effect some changes in the (tangible or intangible) state of the world, typically by an expenditure of effort or energy. Note that it is not required that any tangible object is moved, changed, produced, or destroyed for an action to occur; the effects of an action might be intangible*. In this definition emerges the relevance of the agent or the entity which actually performs the action. KnowRob 2.0 takes the term from the DUL ontology: *An event with at least one agent that is a participant in it, and that executes a task that typically is defined in a plan, workflow, project, etc*. Again, the figure of an agent taking part in the execution of the action is noted. PMK defines the notion of `ActionClass` with three specifications: `Task`, `Sub-task` and `AtomicFunction`. Hence, picking would be an action class whose task can be reachability-test, and it can have a sub-task that provides a list of potential grasping poses. Similarly, in OROSU the notion of sub-task has also been included to form complex actions. CARESSES and OROSU do not provide any natural language definition for `Action`, even though they include the term in their ontology.

Task Most of the studied ontologies agree on the fact that there exists a relationship between `Task` and `Action`. However, as stated in Section 2.3.1, there is not a common agreement on the exact relationship, and each framework/project employs a distinct formalization. In the ROSETTA ontology, `Task` is formalized as a disjoint with: `Operation`, `Skill`, `Physical object`, and `Property`, but no further information is provided. Another example of formalization is found in PMK, where `Task` is a sub-class of `Action`. ORO views it as *an action considered in the specific context of robotics*. KnowRob 2.0 uses DUL's definition: *an event type*

that classifies an action to be executed.

Activity Only CARESSES covers the term of `Activity`, which is formalized as a sub-class of `Entity` and has some sub-classes: `Cooking`, `Reading`, `Sleeping`, etc. Natural language definitions are not provided.

Behavior None of the frameworks defines `Behavior`.

Function Just PMK includes a term related to `Function`, specifically, the notion of `AtomicFunction`, a sub-class of `Action`. It refers to `Function` from a computational point of view (e.g. `Algorithm`). In the other frameworks, it is used the concept of `Algorithm` *to define computational tools* (OROSU) or provide some relations to express the intended (or primary) function of objects (KnowRob 1.0). We consider that in these two last cases, it is not possible to state that the term is covered by the ontologies.

Plan KnowRob 2.0 takes DUL's definition of `Plan`: *a description having an explicit goal, to be achieved by executing the plan*. Each plan defines a task that can be executed by following the plan, however, there may be different plans defining the same task. The execution of a plan is a situation that satisfies the plan, and that defines what particular objects will take what roles when the plan is executed. ORO does not provide a definition in natural language, but from the formalization, `Plan` *is equivalent to a thing with a temporal extent*, which is either a `Situation` or a `Time Interval`. The ORO definition seems to be a bit inconsistent since, surely, a time interval is not a plan in the common sense.

Method The only framework that captures the notion of `Method` is KnowRob 2.0, where the definition provided by DUL is adopted: *A method is a description that defines or uses concepts in order to guide carrying out actions aimed at a solution with respect to a problem*. This notion is similar to the notion of `Plan`, but more general in that variations when following the same method could satisfy different plans.

Capability The ROSETTA ontology defines `Capability` as *a property of a skill* while in PMK the property `has capability` is defined as *a property of a robot*. In KnowRob 1.0, as part of the module SRDL[1], capabilities *are considered to exist when the robot has a component which enable them*. In KnowRob 2.0, `Capability` is formalized as sub-class of `Quality` meaning that

[1]The Semantic Robot Description Language (SRDL) extends KnowRob with representations for robot hardware, robot software and robot capabilities.

agents have individual qualities that may change over time, i.e., due to attrition, new software components being available, etc.

Skill The term of `Skill` is the core of the ROSETTA ontology where *a skill represents an action, that might be performed (by a device) in the context of a production process*. Similarly to the task-action dichotomy, skills are used to classify particular actions that occurred. However, it also implies that the robot or device has the capability to manifest the skill.

Hardware component The specific term of `Hardware component` is not tackled in any of the studied works. However, most of them address one or more concepts related to its notion. The first version of KnowRob included the Semantic Robot Description Language (SRDL), which considers representations for robot hardware, among others, however, KnowRob 2.0 no longer supports it. OROSU, from SUMO ontology, makes use of the term `Device` which *is an artifact whose purpose is to serve as an instrument in a specific subclass of a process*, where `Artifact` *refers to any object that is the product of a making*. A similar definition is found in ORO, where an `Artifact` *is a specialization of inanimate object, and each instance of artifact is an at least partially tangible thing that was intentionally created by an agent partially tangible (or a group of them working together) to serve some purpose or perform some function*. ROSETTA also includes the term of `Device`: *an active physical object which has some skills*. These notions are used to define `Sensors`. In PMK it is possible to find the terms `Actor Class` (e.g. robot components), `Sensor Class` (e.g. device components), and also the term `Artifact`, but none of those terms is defined using natural language. As we can see, definitions of hardware components are closely related to the processes and events in which they play a role.

Software component KnowRob includes the Semantic Robot Description Language (SRDL), which extends KnowRob with representations for robot software, among others. Terms related to the notion of `Computer-based Algorithm` can be found in both OROSU and PMK. ROSETTA ontology defines `Software` as *an abstract which has some skills*.

Interaction The term of `Interaction` is not defined in any of the frameworks we have surveyed. Nevertheless, in PMK it is possible to find the notion of `Interaction Parameter`, which is defined as a data property. Note that this does not mean that the notion of `Interaction` is captured.

Communication In the ROSETTA ontology, `Communication Property` *stands for the description of the communication parameters*. ROSETTA actually includes the term of

`Communication`, but as a subclass of `Device`, which hinders the understanding of it. Therefore, we cannot say that ROSETTA defines the notion of `Communication` which is needed for our domain. Apart from that, the only complete and coherent definition related to `Communication` is found in KnowRob 1.0, where the term of `Communicating` is taken from the Cyc ontology. It *is a specialization of purposeful action* and characterized by one or more information transfer events. Each instance of `Communicating` is an event in which the transfer of information between agents is a focal action; communicating is the main purpose and/or goal of the event.

2.4.2 Comparing the frameworks based on their reasoning scope

In this section, we analyze whether or not the different frameworks are used to support robots to perform the cognitive capabilities presented in Section 2.3.2. Table 2.2 summarizes the results of the analysis, showing the capabilities covered by each of the projects.

Recognition and categorization Ros et al. present a use-case where ORO is used to support object detection by disambiguating incomplete information extracted from human-robot interaction [Ros et al., 2010]. Specifically, the work proposes a scenario in which a human provides vague instructions such as: *look at that object*, where the object can correspond to several entities in the environment. The ontology is used to represent facts about the user's visual spectrum, and the description of objects so that the system is able to infer/recognize which is the most likely object.

Within the framework of CARESSES, Menicatti et al. [Menicatti et al., 2017], introduce an approach for human activity recognition where cultural information (represented using ontologies) drives the learning improving the performance of the classification/categorization. Three human activities are considered: lying on the floor, sleeping on a futon and sleeping on a bed. Specially, lying on the floor and sleeping on a futon are extremely similar classes, thus, cultural knowledge (e.g. user is from Japan), is shown to improve the performance of the recognition algorithm.

KnowRob 2.0 is concerned with acquiring experiential knowledge from observations. One of the considered modalities is the virtual reality where force interactions can be monitored trivially. However, the intention and the task that the human executes might be unknown. KnowRob uses ontologies to represent tasks as patterns of force interactions, and state changes to be able to recognize high-level activities given force event and state observations [Beßler et al., 2023].

Cognitive Capability	KnowRob	ROSETTA	ORO	CARESSES	OROSU	PMK
Recognition and categorization	[Beßler et al., 2023]	-	[Ros et al., 2010]	[Menicatti et al., 2017]	-	-
Decision making and choice	-	-	-	[Bruno et al., 2019b]	-	[Diab et al., 2018], [Diab et al., 2019]
Perception and situation assessment	[Beetz et al., 2015b]	-	[Ros et al., 2010], [Sisbot et al., 2011]	-	-	[Diab et al., 2019]
Prediction and monitoring	[Tenorth and Beetz, 2012]	-	-	-	-	-
Problem solving and planning	[Beßler et al., 2018], [Tenorth and Beetz, 2012]	-	-	-	-	-
Reasoning and belief maintenance	[Beßler et al., 2018], [Tenorth et al., 2010a]	-	[Warnier et al., 2012]	[Bruno et al., 2019b]	-	[Diab et al., 2019]
Execution and action	[Beetz et al., 2010], [Tenorth et al., 2014], [Tenorth et al., 2010b]	[Stenmark et al., 2015]	-	[Sgorbissa et al., 2018]	[Gonçalves and Torres, 2015]	-
Interaction and communication	[Yazdani et al., 2018]	-	[Ros et al., 2010], [Lemaignan et al., 2011]	[Bruno et al., 2018, Bruno et al., 2019b]	-	-
Remembering, reflection and learning	[Beetz et al., 2018], [Beetz et al., 2015c]	[Stenmark et al., 2017], [Topp et al., 2018]	-	-	-	-

Table 2.2: List of cognitive capabilities for the autonomous robotics domain and their coverage in the different chosen frameworks/ontologies. It is possible to find the reference to the articles in which the different reasoning capabilities are addressed using the ontologies.

Decision making and choice In the context of CARESSES, the robot builds a model of a person, which is represented using the ontology. The robot adapts its behavior to the facts of the knowledge base, however, the adaptation to the user is really slow using the ontology. Therefore, it is also used a Bayesian Network for speeding up the adaptation to the person by propagating the effects of acquiring one specific information onto interconnected concepts [Bruno et al.,

2019b].

Diab et al. describe a robot system which adapts the execution of plans with the support of the PMK ontology [Diab et al., 2018, Diab et al., 2019]. Based on the beliefs about the workspace (reachability of objects, feasible actions to execute, etc.), the system makes decisions about the distribution of actions among different robotic arms, and also about action's parameters, slightly modifying the original plan.

Perception and situation assessment A situation assessment reasoner, which generates relations between objects in the environment and agents' capabilities, is presented in the context of ORO [Sisbot et al., 2011]. Being fully integrated into a complete architecture, this reasoner sends the generated symbolic knowledge to a fact base which is built on the basis of an ontology and which is accessible to the entire system. The authors discussed how, based on spatial reasoning and perspective taking, the robot is able to reason from the human's perspective, reaching a better understanding of the human-robot interaction.

Ros et al. present a use-case where ORO is used to disambiguate incomplete information extracted from human-robot interaction [Ros et al., 2010]. Specifically, the work proposes a scenario in which a human provides vague instructions such as: 'look at that object', where the object can correspond to several entities in the working environment. Therefore, the robot is able to assess which is the situation of the environment using the ontology.

One example of a knowledge-based perception system is RoboSherlock [Beetz et al., 2015b]. It uses IBM's UIMA architecture to implement an assembly of perception experts. The problem is that there is no single pipeline that can handle all the different perception tasks, e.g. the pipeline for detecting transparent or shiny objects would be different from the one to detect "regular" objects. RoboSherlock uses KnowRob to define what the different perception experts are, what input they expect, what output they generate, etc. This information, together with background knowledge KnowRob provides, is used for the dynamic composition of perception pipelines.

Finally, in the PMK framework, a tagged-based vision module is proposed [Diab et al., 2019]. In this module, the tags are used to detect the poses and IDs of world entities and assert them to the PMK to build the domain knowledge. Then, a reasoning mechanism is used to provide the reasoning predicates related to perception, object features, geometric reasoning, and situation assessment. Particularly, for situation assessment, an evaluation-based analysis is proposed which generates relations between the agent and the objects in the environment based on the perception outcomes, being these relations used later to facilitate the planning process.

Prediction and monitoring One important aspect for robots that do manipulation tasks is to predict the effects of actions. KnowRob 1.0 introduced the notion of pre- and post-actors of actions [Tenorth and Beetz, 2012]. Pre-actors are the entities that must be known before the robot may execute the action, and post-actors describe what is expected when the action is successfully executed. For example, the task of *cracking an egg* would have a pre-actor of type egg that takes the role of being the *destroyed* entity in the action, while the yolk and the shell would be considered as *created* entities. Hence, the robot can predict what action it needs to execute in order to obtain some egg yolk, e.g. in case egg yolk is required in a recipe that the robot tries to cook. In KnowRob 2.0, the pre- and post-actor relations are replaced by corresponding role concepts describing roles that need to be taken by some entity when an action is performed.

Problem solving and planning KnowRob 2.0 uses ontologies for dynamic plan generation. This has been elaborated with respect to assembly tasks [Beßler et al., 2018]. The rationale is that the goal state, a fully assembled product, is described in an ontology, and that the robot compares its belief state with the goal state in order to infer what steps are required, and what objects are missing to build the product from parts that are available. KnowRob 1.0, on the other hand, used action definitions axiomatized by roles that are separated into input and output of the action, and, in addition, defined a partial ordering on steps of a task [Tenorth and Beetz, 2012]. This information was used to generate possible sequences of steps that would execute a task such as making pancakes.

There are some works in which the knowledge used to generate a plan is represented using an ontology. However, unlike the other works presented above, the ontology is not directly used to generate the plan, it only complements the planning. Therefore, we exclude those works from Table 2.2, but it is worth mentioning them. For instance, KnowRob 1.0 has been used to represent motion constraints that were used by a constraint-based motion planner to generate appropriate motions for the task ahead, and the objects involved [Tenorth et al., 2014]. Another example is PMK [Diab et al., 2019], which can serve as a tool for any planner to reason about task and motion planning inference requirements, such as robot capabilities, action constraints, action feasibility, and manipulation behaviors.

Reasoning and belief maintenance As we are only considering frameworks that use ontologies, one can also expect that some form of reasoning is supported. Be it via a standard reasoner, or by some custom rules that can infer new facts from given ones. For example, PMK and KnowRob both use predicate logic rules to work with knowledge encoded in ontologies. However, belief maintenance is not covered by all considered systems.

Regarding ORO, Warnier et al. propose a novel algorithm for belief maintenance, which relies on the use of the ontology to represent the environment's facts [Warnier et al., 2012]. The robot builds an individual symbolic belief state for each agent participating in the task.

Within the CARESSES project, Bruno et al. [Bruno et al., 2019b], present an algorithm for belief maintenance of person-specific knowledge, using cultural knowledge to drive the search.

Diab et al. [Diab et al., 2019], propose to use the PMK ontology to generate semantic maps of the robot's workspace enhancing its belief maintenance. By means of computer vision methods, the robot detects objects and their properties (e.g. poses) and, using the ontology, it stores a symbolic representation of the workspace. A reasoning process over those symbolic beliefs allows to make assumptions about abstract spatial relations (e.g. cup *on* the table).

KnowRob 2.0 also maintains a belief state [Beßler et al., 2018], however, there is no detailed documentation about how the belief state is maintained, and how the system would cope with contradictory information. However, a notion of `SemanticMap` exists in the KnowRob ontology, and it has been used in KnowRob 1.0 to build environment representations that include spatial information and encyclopedic information about objects in the map [Tenorth et al., 2010a].

Execution and action In order to enhance robot autonomy when executing actions, ROSETTA proposes a system that translates high-level task-oriented language (ontology-based) into either the robot's native code, or calls at the level of a common API like, e.g., ROS, or both. This system is capable of handling complex, sensor-based actions, likewise the usual movement primitives [Stenmark et al., 2015].

Related to CARESSES, Sgorbissa et al. discuss how guidelines describing culturally competent assistive behaviors can be encoded in a robot to effectively tune its actions, gestures and words [Sgorbissa et al., 2018]. In the same context, an online constraint-based Planner is used together with the cultural knowledge base to adapt the execution of the robot's actions [Khaliq et al., 2018]. When launched, the planner requests operators and actions from the Cultural Knowledge Base. During execution it listens for new goals, updates on the execution status of actions, and messages about the state of the environment and people in the environment.

Gonçalves et al. discussed how the use of the OROSU ontology is beneficial to track the execution of actions of robotics systems in medical (surgical) scenarios [Gonçalves and Torres, 2015]. In this work, the main purpose is to adapt the robot pose to possible unexpected motions while performing drilling tasks during surgery. The robot pose adaptation is performed following the approach presented in [Torres et al., 2015]. The overall process is modeled with the OROSU ontology, which controls the robot's actions and sub-actions and allows the user to follow the sequence of those actions.

The *Cognitive Robot Abstract Machine* (CRAM) is a plan executive that is grounding abstract plan descriptions such that they become executable by robots [Beetz et al., 2010]. To find suitable task instantiations, CRAM uses KnowRob to, e.g. query for objects in the belief state, where they are located, how they can be operated, etc. The queries to the knowledge base are explicit steps in the plan instantiation procedure of CRAM. Nowadays, CRAM has switched to the second generation of KnowRob. KnowRob 1.0 has further been used to ontologically describe motion constraints that are used by a constrained-based motion controller to generate motions that execute a specific task [Tenorth et al., 2014]. KnowRob 1.0 has also been used to transform vague task descriptions in natural language from the Internet to an ontological representation using WordNet to disambiguate word senses [Tenorth et al., 2010b]. Finally, the ontological representation is mapped into the robot's plan language such that the robot can execute the task.

Interaction and communication Lemaignan et al. present a simple natural language processor which employs ORO to allow robots to dialog with humans [Lemaignan et al., 2011]. The robot parses English sentences and, by means of the knowledge base, infers the sense of the sentences and answers the human (both in English and with RDF statements). Again using the ORO ontology, in a scenario where a robot disambiguates the information provided by the user [Ros et al., 2010], the ontology triggers the robot-human interaction (e.g. asking the user for further information).

In the scope of CARESSES, Bruno et al. describe two scenarios where human-robot speech-based interaction is adaptable by means of cultural knowledge-based assumptions [Bruno et al., 2018, Bruno et al., 2019b]. The system stores knowledge about the cultural information of the users, which is used by the robot's finite state machine to control the interaction.

KnowRob 2.0 was used and extended in a research project that was concerned with mixed human-robot rescue tasks [Yazdani et al., 2018]. The scenario is that a team of different robots has to locate an avalanche victim in hilly terrain where, first, a flying robot scans the area, and then, after the victim is found, the robot communicates the particular location, and an image captured by its camera to the human operator, and to the other robots. KnowRob's ontology is used to represent the communication acts. However, the communication was not natural but was using a custom protocol.

Remembering, reflection, and learning Different approaches combining ontologies and robot learning are proposed in the context of ROSETTA. First, ontologies are used to support kinesthetic teaching so that the learned primitives are semantically represented as skills [Stenmark et al., 2017]. Second, Topp et al. discuss how the representation of already

learned robot skills enhances the transfer of knowledge from one robot to others [Topp et al., 2018].

KnowRob 2.0 introduces the notion of *narrative-enabled episodic memories* (NEEMs) [Beetz et al., 2018]. Each time a robot performs a task, a detailed story about the activity is stored. The story includes a narrative represented in an ontology that describes what events occurred, when they occurred, and what objects play what roles in the events. The narrative is coupled with control-level data such that learning mechanisms can correlate parts of the narrative to the control-level data that was monitored during execution. Earlier, in the context of KnowRob 1.0, the knowledge web service openEASE was introduced [Beetz et al., 2015c]. openEASE is used as a central storage for experiential knowledge, and it has been adopted for KnowRob 2.0 as a storage for NEEMs.

2.4.3 Comparing the frameworks based on their application domain scope

In order to finish the comparison of the different works, in this section, we discuss in which domain they have been applied. Recall that, following the classification proposed in Section 2.3.3, two main domains are considered: industrial and service robotics. Along this section, we will specialize those upper-level domains into the more specific sub-domains where the frameworks were used.

In principle, it is noticed that most of the frameworks were conceived to be used in service scenarios. Indeed, the only framework that is intended to be used in industrial scenarios is ROSETTA, whose case studies solve industrial problems such as: human-friendly robot programming, safe human-robot interaction, etc. Moving to the frameworks focused on service robotics, ORO proposes case studies where the robot is meant to perform everyday activities which usually take place in houses or similar environments such as human activity recognition, human-robot speech interaction, etc. Closely related to it, we find the case of KnowRob, which is used in the framework of household scenarios, but also in scenarios where the robot is expected to perform some professional service tasks (e.g. cooking, in-store logistic processes, etc.). PMK presents case studies where the main objective is to enhance robot manipulation, which is a general-purpose robot ability that could potentially be used in a wide range of scenarios. Nevertheless, PMK has not been used nor thought to be used in industrial scenarios, thus, we can consider it behind the umbrella of the service robotics domain. CARESSES is entirely developed for the assistance of elder people by means of autonomous robots with cultural-related knowledge. Finally, OROSU is mainly applied to the medical domain, particularly, the surgical robotics sub-domain. In Table 2.3, we provide a summary of the previous discussion.

Framework	KnowRob	ROSETTA	ORO	CARESSES	OROSU	PMK
Application Domain	Service	Industrial	Service	Service	Service	Service

Table 2.3: Application domain for each chosen work.

2.5 Discussion

The first classification criteria proposed in our work concerns the ontological scope of the projects, namely which terms are covered by the projects' ontologies (see Section 2.3.1). Looking at Table 2.1, we can see how most of the projects include the terms `Action` and `Task` to capture the notion of robots acting and causing changes in their environment. In the same table, we observe that `Behavior`, `Function` and `Method` are only rarely covered. This is rather surprising given the fact that they are extremely related to applications where agents execute actions. We think that this is the case due to their polysemous nature, therefore, more work is needed in order to come up with a standard definition for them. Some other terms such as `Plan`, `Capability`, `Hardware` and `Software` are defined by at least half of the surveyed projects, which indicates that they are relevant for the domain but still it is necessary to continue working on them. We also discovered that the terms `Interaction` and `Communication` are not defined in any of the projects, even though some projects propose scenarios where robots interact and communicate with other agents (e.g. humans). Hence, no reasoning is done in this regard, and it would be helpful to continue exploring human-robot interactive scenarios (e.g. collaborative or assistive) to formalize the domain knowledge. Surprisingly, some other terms with strong connotations in robotics such as `Activity` and `Skill` are only rarely considered in formal models. In general, we have identified the existence of inconsistencies and different points of view from one framework to another. Therefore, we believe that it is necessary to work towards an agreement on the definition of the relevant terms in our domain.

Regarding the second classification criteria (see Section 2.3.2), it was analyzed which cognitive capabilities of autonomous robots have already been supported by the ontologies of each of the projects. Our review has shown that a wide range of cognitive capabilities are already covered, at least in a prototypical way, by ontology-based approaches (see Table 2.2). Of the nine cognitive capabilities considered in this work, two are commonly tackled within the studied frameworks: reasoning and belief maintenance, and execution and action. Reasoning and belief maintenance are well supported by many ontology formalisms as standard reasoners exist that can perform these tasks automatically. On the other hand, robots are essentially developed to automatize the execution of actions; hence, it is understandable to find several

works tackling this task. Surprisingly, the literature falls short of comprehensively addressing some important cognitive capabilities. For instance, just a few works discussed the use of ontologies to support recognition and categorization. We defend that more research should be done toward this, because the inference power of ontologies would be a great tool to recognize and categorize different robot's experiences (e.g. collaborative and adaptive events). There is also a lack of works discussing the use of ontologies for robotic decision making, problem solving and planning. Probably, the existence of other widely used formalisms to do planning (e.g. PDDL) is the principal reason which partially explains this fact. However, we think that ontologies for robots should have more presence in the decision making and choice literature. Robot ontology-based decisions would certainly be sound, and ontologies might be a great abstraction for modeling decisions that are more general than those made in plans. Furthermore, a formal representation of robots' decisions would play an important role in explainable robotics. In this regard, the review also revealed that ontologies have been used for robot learning tasks and to store robot memories. Those contributions might seed some light on some of the issues that arose from the current trend of using non-explainable machine learning approaches. However, there are no specific works that focused on ontology-based robot's reflection and explanation generation, thus, further research shall be conducted in this direction aiming for trustworthy robots. These conclusions influenced our research, thus this thesis presents contributions regarding ontology-based approaches for: recognition of robot experiences, robot decision making and explainable robotics.

Lastly, we also studied the application domain of the selected projects. Most of the projects were conceived with the purpose of being used in service robotics applications. Indeed, just ROSETTA was specifically designed for industrial robotics applications. It is true that the rest of the frameworks could be adapted to be useful in industrial settings too. However, we think that putting the focus on those industrial applications would translate into the formalization of domain knowledge that is not covered in the literature (e.g. collaborative events, safety issues, etc.). Hence, this thesis addresses the challenge of modeling the domain knowledge of collaborative industrial tasks, of course, aiming for conceptualizations that are as general and reusable as possible.

Considering the work presented in this chapter, it is possible to state that ontologies have proved to be valuable for the robotics domain in order to support robot autonomy. It is true, however, that the great effort made by all the frameworks discussed in this chapter should be continued and extended with new applications. Furthermore, it is still pending to promote the reuse of existing ontologies, which seeks for homogeneity and interchangeability among different frameworks. This will only be possible if researchers share and properly document their

contributions. In this regard, we have summarized the content of our review on a web page[2] where users can access the major findings of our work. Our aim is to continuously maintain and improve this page to provide researchers and ontology users easy access to related work. Specifically, the page allows to search/select by projects and by each of the three scopes proposed in our work.

As a final remark, we consider that all the projects analyzed in this chapter provide enough information to allow to reuse and reproduce their results. However, KnowRob is by far the framework with the best existing documentation, including: code, ontologies, as well as wiki pages explaining how to install and to use all its different tools. Hence, we decided to use it as the general knowledge representation and reasoning framework for the ontology-related works presented in this thesis. Note that this obviously conditioned some of the decisions made during the ontological modeling (e.g. selection of an upper-level ontology, used formal language, etc.).

[2] https://ease-crc.org/ontology-survey-2019

chapter three

Inferring the intentions of humans in collaborative experiences

> *..ogni forma di vita poggia su movimenti intenzionali aventi uno scopo non soltanto in se stessi..*
>
> **— Maria Montessori**
> (La mente del bambino: Mente assorbente)

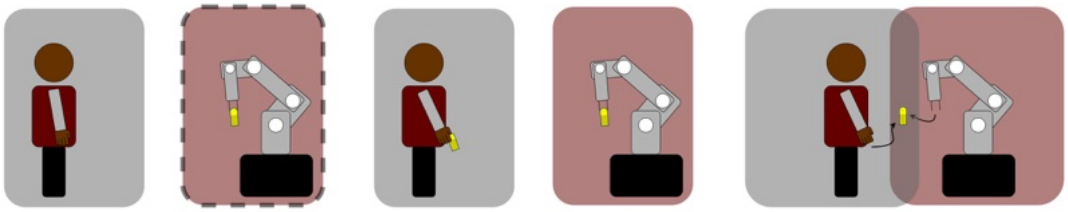
The literature review discussed in Chapter 2 suggests analyzing and conceptualizing the knowledge of interest from human-robot collaborative scenarios. It was also discovered that it would be especially convenient to explore the use of ontological conceptualizations to support robots' cognitive capabilities such as decision making and choice, and recognition and categorization. We will start by implementing a real human-robot collaboration example to gain first-hand experience. This will be used to identify reality patterns from the robots' perspective that need to be conceptualized. In a collaborative context, certainly one of the most relevant aspects for robots to recognize is the intention of their human collaborators, which can be used during robots' decision making. Hence, this chapter investigates the development of a robotic system to recognize and categorize the intention of humans for later robot adaptation. The work contributed with a novel dataset of physical human-robot interaction, and two methods for human intent recognition that were trained and evaluated using the dataset. It was also evaluated how well the system generalized to new users (N=15), in a scenario inspired by a realistic industrial application. By the end of the chapter, it is also discussed how this work helped framing the scope of the conceptualized ontological models presented in this

thesis. For instance, the research revealed that there might be different types of collaboration, also that when the intention was not properly recognized, the collaborative event might be interrupted, because the robot and the human no longer had a shared intention (i.e. plan). Indeed, when such misalignment occurred, informing users about the robot's inference was found to be useful, which is aligned with the idea of maintaining a mutual and shared understanding and the need for explainable robots.

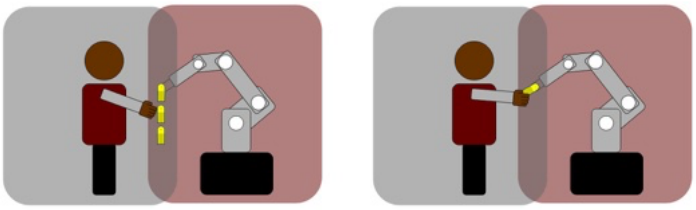
3.1 Motive

In the last years, the figure of *Collaborative Robots* or *Cobots* has emerged [Michalos et al., 2014, Tsarouchi et al., 2017, Wang et al., 2019]. These robots are specifically designed for direct interaction with a human within a defined collaborative workspace [Roy and Edan, 2020]. Note that meaningful human-robot collaboration requires freeing robots from their work cells and putting them closer to operators, possibly compromising human safety [Michalos et al., 2015, Villani et al., 2018]. In this regard, collaborative robots have meant great progress towards a safe coexistence of operators and industrial robots. Nevertheless, scenarios where humans and robots closely share the space and the execution of a task require the use of robots equipped with complex cognitive capabilities [Someshwar and Edan, 2017].

This thesis aims to explore the relevant knowledge involved in cognitive robot capabilities (e.g. recognition or decision making) in collaborative scenarios. Intuitively, one might observe a relationship between the complexity of those capabilities and how advanced the collaboration is. Indeed, Bauer et al. [Bauer et al., 2016] proposed a taxonomy of five levels of cooperation between robots and humans (see Fig. 3.1). The authors stated that most of the current real applications of industrial robots are based on the cooperation levels *coexistence* and *synchronized* [Someshwar et al., 2012, Someshwar and Kerner, 2013]. It seems reasonable to think that the most advanced levels will offer a richer exploration to our purposes. Hence, motivated by the scarcity of applications where more complex levels of cooperation are addressed, this chapter discusses an approach focused on a scenario corresponding to the fifth level, *collaboration*. This hands-on experience serves two purposes: aiding in the framing of the ontological scope of the models presented in this thesis and advancing the state-of-the-art in human-robot close collaboration. Fig. 3.2 depicts the proposed setup, where a human and a robot exchange forces while sharing the execution of a task inspired by a realistic industrial scenario.



(a) Cell. (b) Coexistence. (c) Synchronized.

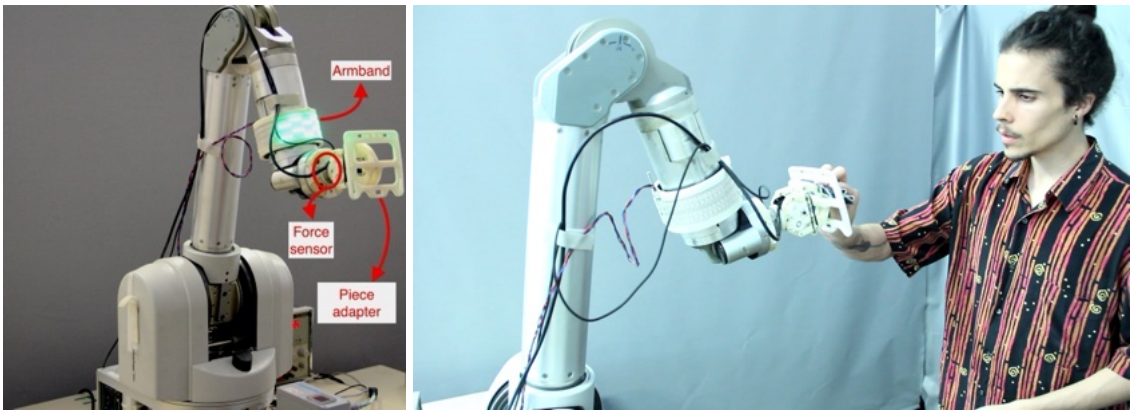


(d) Cooperation. (e) Collaboration.

Figure 3.1: Human-robot cooperation levels in industrial environments. (a) The level *cell* involves no collaboration at all, the robot remains held inside a work cell. (b) *Coexistence* removes the cell but humans and robots do not share the workspace yet. (c) *Synchronized* allows the sharing of the workspace but never at the same time, humans and robots operate in a synchronized manner. (d) In the level *cooperation* the task and the workspace are shared, but humans and robots do not physically interact. (e) The level *collaboration* considers full collaboration where operators and robots exchange forces.

3.2 Related work

The literature contains multiple data sets for human-robot interaction, although most of them focused on social robotics scenarios, where the most common mean of interaction is not physical but verbal. Those data sets usually contain video, speech (audio and transcripts), robot joint-sate, physiological data (e.g. bio-signals) or subjective data in the form of questionnaires [Mohammad et al., 2008, Jayagopi et al., 2013, Bastianelli et al., 2014, Celiktutan et al., 2019, Webb et al., 2023]. There are also some data sets capturing robot force/torque data, but extracted from scenarios in which robots and humans do not interact. Yu et al. [Yu et al., 2016], presented a dataset in the context of pushing tasks where a robot pushes an object along a specific surface. For each combination of an object's shape and a surface's material, the dataset contained forces measured in the pusher and the poses of both the object and the pusher. Another interesting dataset involving forces was introduced by De Magistris et al. [Magistris et al., 2018], where authors recorded a force-signal dataset used to



(a) Robot hardware components. (b) User pose with respect to the robot.

Figure 3.2: Proposed scenario inspired by an industrial collaborative robotic task in which the robot adapts its state to the human's intention. (a) The force sensor is used to infer the human's intention, the armband is used to inform the user about the robot's internal state, and the piece adapter eases the grasping of the object. (b) While the robot holds the object, the human performs a frontal polishing of it. Three different operator intents are considered: polishing, moving the robot to another pose, and grabbing the object.

learn peg-in-hole robot tasks. The dataset comprised force-torque and pose information for multiple variations of convex-shaped pegs. Huang et al. [Huang and Sun, 2018], introduced a dataset containing force/torque signals and poses of an end effector tool. Data was recorded from humans performing a set of different motions making use of the same tool that the robot would use, enabling the transference of knowledge. Those works were great advances toward the development of statistical algorithms that can effectively generalize and thus perform robotic tasks without explicit instructions. In this regard, it would be great if collaborative robots had the ability to generalize to different users and situations when performing tasks that require exchanging forces with humans, thus, a novel dataset is proposed in this work.

There are several research works discussing applications where humans and robots physically interact. However, in many of the cases, the force exchange between humans and robots is ignored or undesired. Cherubini et al. [Cherubini et al., 2016], discussed a collaborative scenario where a human and a robot shared the task of *rzeppa homokinetic* joint insertion. In their proposal, the robot just remained stiff, and the force-based information was not used in the decision making of the robot (e.g. to adapt its state). Maurtua et al. [Maurtua et al., 2017], conducted a set of experiments aimed at measuring the trust of workers on fence-less human-robot industrial collaborative applications. In all their experiments, the force was undesired, thus the robot stopped when an external force was detected. De Gea Fernández et al. [de Gea Fernández et al., 2017], described an industrial situation in which two robotic

arms collaborate with an operator. Both robots avoided any physical interaction with the human as long as possible, and when a physical interaction occurred, they remained in a compliant mode, thus the force was ignored. Raiola et al. [Raiola et al., 2018], addressed the problem of learning virtual guiding fixtures, analogous to the use of a rule when drawing, in human-robot collaboration. Even though there was physical interaction during the task execution, the robot did not react to the force while guiding the human. In the work presented by Munzer et al. [Munzer et al., 2018], a human and a robot performed sub-tasks of a shared task: wooden box assembling. The robot and the human shared forces and the robot was able to adapt to the situation, but not using the force, just using vision, or being explicitly asked to do it by voice commands or instructions using a graphical interface.

Closer to our research, the literature also shows some recent works in which robots adapt their behavior based on the physical interaction between humans and robots. Losey et al. [Losey et al., 2018], conducted a comprehensive review of intent detection and other aspects within the context of shared control for physical human–robot interaction. It is especially interesting how the paper was structured, talking about three aspects covered in our work: user intent recognition, shared control between humans and robots, and methods to inform the human operator about the robot's state. Peternel et al. [Peternel et al., 2018], estimated human fatigue to adapt how much a robot was helping in human–robot collaborative manipulation tasks: sawing and surface polishing. Rozo et al. [Rozo et al., 2016], proposed a framework for a user to teach a robot collaborative skills from demonstrations. Their approach combined probabilistic learning, dynamical systems, and stiffness estimation to encode the robot's behavior along with the task. Hence, the method allowed a robot to learn not only trajectory-following skills, but also impedance behaviors. Those two works focused on robot adaptation at the low-level control, while in our research the focus is on adaptation at the symbolic level of the task. Mazhar et al. [Mazhar et al., 2018], proposed a scenario where a human and a robot physically interacted through a handover of an object. Force signals were used to identify different phases of the sequence of actions. For instance, when a force threshold was exceeded, the system interpreted that the robotic hand should close to grasp the object. Zhao et al. [Zhao et al., 2018], presented an operator's intention recognition approach inspired by a collaborative sealant task. The intentions, rather similar to ours, are also used to adapt the state of the robot, just as in our work. However, the interactions they proposed are simplistic as the classes could be discriminated by using force thresholds. The novel dataset recorded for our work not only includes simple *mechanical* movements, but also more *natural* human-robot interactions. Finally, Gaz et al. [Gaz et al., 2018], introduced a new robot control algorithm for a collaborative polishing task, quite similar to the one discussed in this chapter. However, there are major differences between our works. First, they only considered two

intentions and thus two robot modes: stiffness (while polishing), and compliance (while modifying the end effector orientation). Second, they did not propose a learning method to recognize the human's intentions, the cases were differentiated because the human applied force to different parts of the robot.

3.3 Force-based dataset of physical human-robot interaction

In this section, we provide all the relevant information related to the novel dataset [Olivares-Alarcos et al., 2019b] that is used along the evaluations presented in this work. The dataset consists of force/torque signals resulting from the physical human-robot interaction during the performance of a collaborative task consisting on polishing a piece. The dataset is geared to teach robots to identify and predict humans' intentions during the shared task.

3.3.1 The target industrial collaborative robotic scenario

In this work, it is considered a realistic industrial scenario inspired by a manufacturing line of car emblems. The focus is on one sub-process where the emblems are to be coated, and they must be totally clean and polished. Nowadays, the plant operator picks, inspects, and polishes the emblems, to finally place them into another location where they are coated. The objective is that a robot and the human share the task collaboratively. For this work, that process is re-designed so the robot is in charge of the picking and placing tasks, while the operator still inspects and polishes the emblem. Hence, once the robot has posed the piece in front of the operator, the human can perform different actions over the emblem while the robot should infer those actions and adapt to them. Note that the principle mean of human-robot interaction is force-based. The interaction should be natural for the human and the reaction time of the robot should ensure a fluent and efficient collaboration. Note that it is not within the scope of this work to tackle how the robot grasps and places the emblems. Instead, we focus on how the robot, while offering the emblem, can infer the operator's intent and adapt accordingly.

3.3.2 Human intentions and robot adaptation states

Recall that three different operator intents are considered: (a) polishing, (b) moving the robot and (c) grabbing the object. Analogously, there are three different states of the robot: (a) increasing stiffness (named *'hold'*, from now on), (b) decreasing stiffness (*'move'*) and (c) releasing the object (*'open gripper'*), respectively. In the first case, the operator should be able

to do the main objective of the task, polishing the emblem. When applying this sort of force, the robot should be stiff. Otherwise, the polishing action would not succeed. The second operator's intent is regarding ergonomics in industrial scenarios. The operator could get tired of polishing the pieces at the same pose, or there could be another operator with different corporal dimensions and/or abilities. Hence, this time, the force should be done to move the robot to a more comfortable pose. Finally, we also contemplate the case in which the human wants to grab the object (emblem), pulling it from the robot's gripper. In this case, the robot should open the gripper to release the piece.

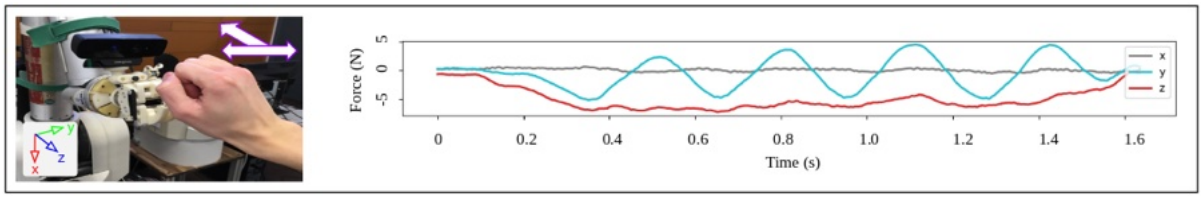
3.3.3 Specifications of the novel dataset

The dataset was recorded using an ATI Multi-Axis Force/Torque Sensor Mini40 SI-20-1, which was fastened to the wrist of the robot, the basis of the end effector (see Fig. 3.2a). We used the default configuration of the sensor, and the measurements were taken at a frequency of 500 Hz.

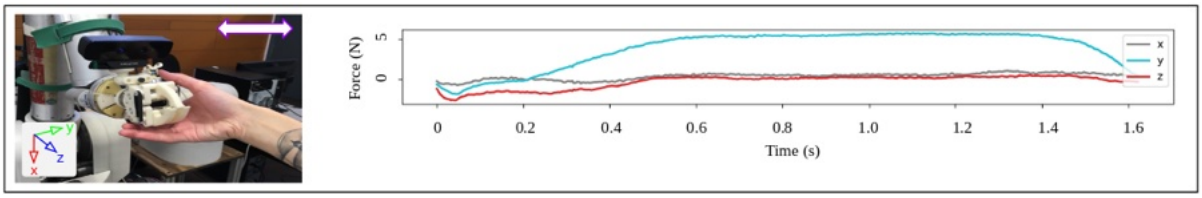
Every sample contains a single sort of interaction, from the beginning to the end of the physical contact. Note that gathered data samples were not of the same length, ranging from half a second to three seconds long. In the dataset, the shorter samples are padded with zero values at the end of the temporal sequences so that all of them have the same length. The dataset contains six different files per each of the three classes, which correspond to the six axes of the force sensor. Each file is named using the force/torque axis and the class label, hence, users can read the samples included in each file and label them appropriately.

The aim is to infer force-based human intentions from *natural* and therefore ambiguous human-robot interactions, but we also evaluated our method with easily distinguishable *mechanical* interactions. In the *mechanical* dataset, each class follows distinct movement patterns, which produce completely different force signals that are easy to discriminate. Meanwhile, in the *natural* dataset, the movement patterns of the classes are much more similar to each other, which makes the classification more challenging. From alternative machine learning approaches, one is selected (see Sec. 3.5.4), and its performance is later evaluated using each of the two datasets in Section 3.5.5. Note that for the evaluation with users, only the *natural* data was used for training.

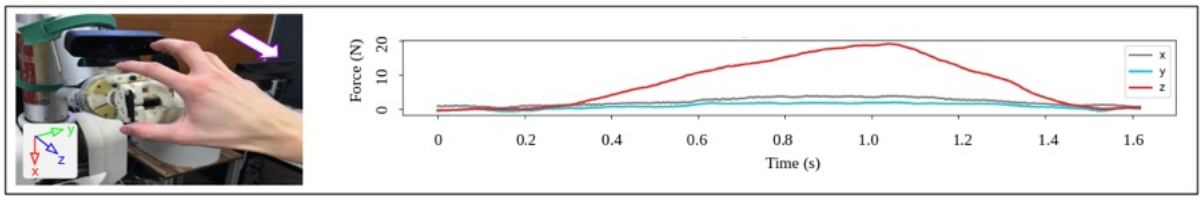
Since it is expected to be easier to classify, the *mechanical* dataset only contains 600 samples. Recall we have three classes and we have used two users, thus, each user did 100 samples of each class. The physical contact was always done following restricted patterns for each intention/class. Fig. 3.3 depicts both, the different axes in which the operator is supposed to apply the force and the corresponding force signals we detect using the sensor. For the polishing intention, we move periodically only in the axis Y and we push towards the robot,



(a) Polish



(b) Move

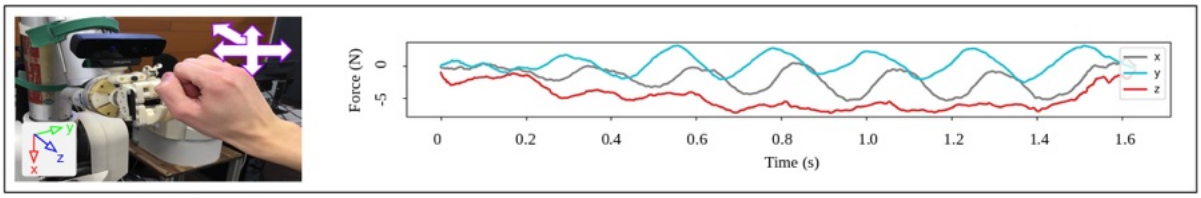


(c) Grab

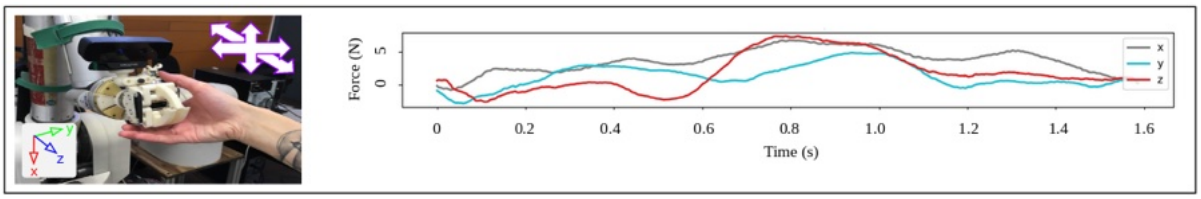
Figure 3.3: *Mechanical* dataset. Human movement patterns (left side) and appearance of the force signals produced by those patterns (right side). Observe how each class (a, b, c) is quite distinguishable from the rest even after only 0.4 seconds. Making use of this dataset to train a model would allow predicting fast with enough confidence. Nevertheless, the movement patterns of the user would be too restricted and the human-robot interaction would not be natural.

negative Z-axis (Fig. 3.3a). In order to move the robot, we move just in one direction for each sample and only in the axis Y (Fig. 3.3b). Finally, to grab the object, we pull the robot's end-effector towards ourselves, positive Z-axis (Fig. 3.3c).

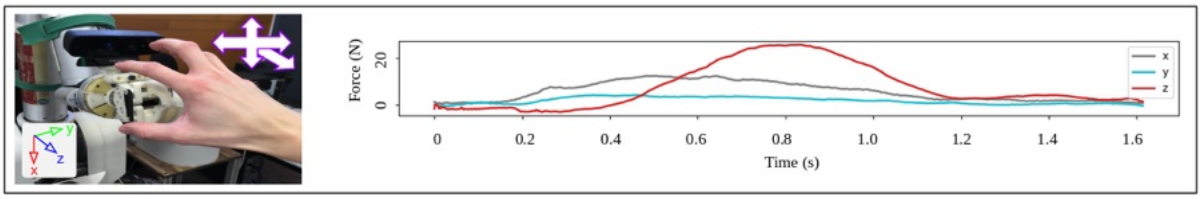
Unlike with the *mechanical* dataset, the *natural* dataset contains more samples, 900. Recall we have three classes and we have used two users, thus, each user did 150 samples of each class. In this case, the physical contact for each intention/class can be done following several natural patterns, which increases the ambiguity between classes. In Fig. 3.4, it is possible to see the different axes in which the operator is supposed to apply the force and the corresponding force signals detected with the sensor. For instance, the intention of polishing can now be done by describing circles and also using the axis X (Fig. 3.4a). The patterns to move the robot now include any of the directions of the three spatial axes (Fig. 3.4b). Finally, the operator could now try to grab the object pulling but not only towards the exact direction of the Z-axis (Fig. 3.4c).



(a) Polish



(b) Move



(c) Grab

Figure 3.4: *Natural* dataset. Human movement patterns (left side) and appearance of the force signals produced by those patterns (right side). Observe how each class (a, b, c) is still similar to the rest even after 0.4 seconds. Due to the richness in movements, a model trained with this dataset would allow a natural human-robot interaction.

Please, recall that, although for illustrative purposes figures only show the linear forces, our classification process uses both torque and linear signals. Together with the dataset, we also provide some Python notebooks which run our proposed approaches using the data [Olivares-Alarcos et al., 2019b].

3.4 Approaches to force-based operator's intent recognition

In order to infer the humans' intentions, two different approaches were proposed and implemented. Both were evaluated on the *natural* dataset, and they were compared to identify the best option considering the objectives of this work. The selected approach was used during the validation carried out in Section 3.6, and also to analyze the differences between the

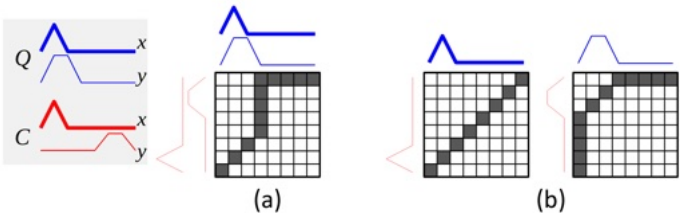


Figure 3.5: Dynamic Time Warping (DTW) for multi-dimensional time series: *dependent* (a) and *independent* (b) DTW. The former consists in computing the DTW similarity path of both dimensions (axis) at the same time. The latter is much simpler, normal DTW is computed separately on each dimension, and their results added posteriorly.

natural and *mechanic* data sets. These results are part of the experimental findings presented in this chapter. Recall that this work seeks a natural human-robot interaction, fast reaction of the robot, and, if possible, an approach that deals with heterogeneous industrial contextual data and knowledge.

3.4.1 Raw-data-based recognition approach

In this approach, using directly the data obtained from the sensor, the classification is done through a k-Nearest Neighbors (kNN) classifier with Dynamic Time Warping (DTW) [Berndt and Clifford, 1994] as metric. Particularly, we have used $k = 1$. While being a simple method, 1NN+DTW performance seems to be hard to beat by other approaches in time series classification problems [Bagnall et al., 2017].

Dynamic Time Warping is a time-dependent algorithm used to measure similarity between two temporal sequences which may vary in speed. For instance, similarities in polishing could be detected using DTW, even if the operator polishes faster or slower than in other cases. DTW is a computational intense technique, with quadratic time and memory complexity. However, there are some ways to accelerate computation, thus in this work, it was used the library Fast DTW [Salvador and Chan, 2007]. DTW was meant to be utilized for uni-variate time series, which is not the case in this chapter, since there are six sensor axes. In the literature, exist at least two obvious approaches to tackle this and generalize DTW for multi-dimensional time series: *dependent* and *independent* DTW (see Fig. 3.5) [Shokoohi-Yekta et al., 2017]. kNN classifier is taken from *scikit learn* library[1]. The implementations of kNN and Fast DTW do not allow to work with multi-dimensional time series, thus it was necessary to adapt the libraries. Apart from those modifications, the default values were used.

[1] https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html

3.4.2 Feature-based recognition approach

In this section, it is proposed a twofold machine learning approach to infer the human operator's intentions. First, the dimensionality of the data is reduced using an unsupervised method: Gaussian Process Latent Variable Model (GPLVM) [Lawrence, 2003]. Then, a Support Vector Machine (SVM) classifier is trained using the lower-dimensional representation of the data. GPLVM is a non-linear dimensionality reduction method that can be considered as a multiple-output Gaussian process regression model where only the output data are given. The inputs are unobserved and treated as latent variables, however, instead of integrating out the latent variables, they are optimized. By doing this, the model gets more tractable, and some theoretical grounding for the approach is given by the fact that the model can be seen as a non-linear extension of the linear Probabilistic Principal Component Analysis (PPCA) [Tipping and Bishop, 1999]. Note that in this case, the temporal sequences are just considered as long feature vectors so that it is not explicitly considered the temporal relation between subsequent signal measurements. However, dimensionality reduction has proved to be an effective technique in time series analysis, in which data is remarkably high dimensional [Su et al., 2018, Villalobos et al., 2018, Seifert et al., 2018].

The implementation of the proposed method, GPLVM+SVM, relies on two existing libraries: *GPy* library[2] for the dimensionality reduction and the *scikit learn* library for the SVM classifier[3]. In the case of the latter, the default values were used for all the parameters. However, concerning GPLVM, it was necessary to set some parameters: kernel, optimizer, and the maximum number of optimization steps. First, the chosen kernel was a combination of the Radial Basis Function (RBF) kernel together with a *bias* kernel. RBF kernel was selected because it is one of the most well-known kernels for non-linear problems. The *bias* kernel was added to enable the kernel function to be computed not only in the origin of coordinates. Second, for the optimization process, it was used one of the optimizers already implemented in *GPy*, limited-memory Broyden–Fletcher–Goldfarb–Shanno [Liu and Nocedal, 1989]. Unlike others included in the library, it was quite stable concerning the number of optimization steps needed to converge, which is why it was selected. Finally, the maximum number of optimization steps was set to 5000, which in most cases was enough for the optimization to converge.

The implementation of the GPLVM algorithm included two different types of latent variable inference: with optimization step (GPLVM-op) and without the optimization step (GPLVM). For this work, the most relevant difference between them was that the inference with optimization would take more time, but it would lead to more accurate results. Nevertheless, as it is shown

[2] https://sheffieldml.github.io/GPy/
[3] https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html

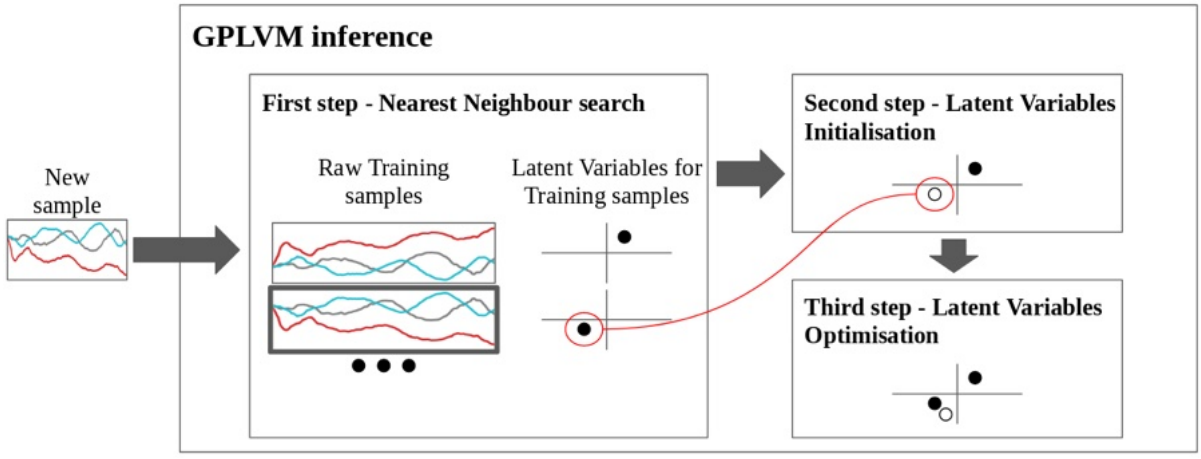


Figure 3.6: Global GPLVM inference process of the latent variables given a new sample in the higher dimensional space. First, the most similar training sample to the new sample is found using Euclidean distance. Second, the value of the latent variables of the most similar training sample (black dot in the first step) is used to initialized the inferred value (see the white dot in the second step). Third, the GPLVM is optimised considering the new sample, which results in a refinement of the inferred latent variables. GPLVM with optimization includes the three steps, GPLVM without optimization stops after the second.

in Section 3.5.3, the inference with optimization did not always ensure better performance. Regarding the inference process, given an already optimized GPLVM and a new sample to infer its latent variables, the inference process is divided into three steps (see Fig. 3.6). The first step, *nearest neighbor search*, is focused on finding which of the training samples is the most similar to the new sample. This is done computing the similarity between the new sample and all the training samples employing the Euclidean distance. The second step, *latent variables initialization*, consists in setting the value of the inferred latent variables to the values of the latent variables of the nearest neighbour found in the previous step. Finally, during the third step, *latent variables optimization*, the value of the initialized latent variables is refined through optimization.

3.5 Evaluation of the force-based operator's intent recognition approaches

3.5.1 Evaluation setup for the proposed approaches

For the validation, it has been applied cross-validation without replacement ten times, the *natural* data was randomly split into training (75%) and test (25%). The chosen metric to evaluate

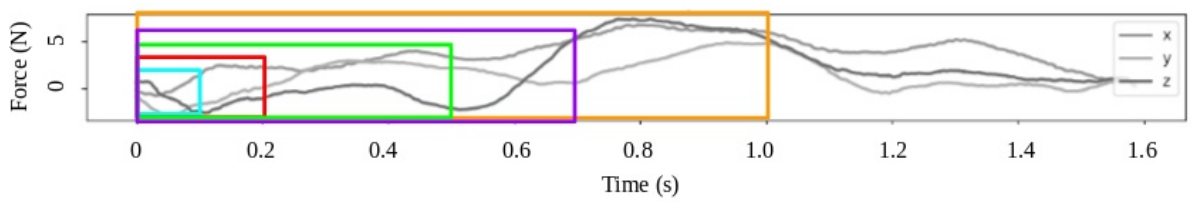


Figure 3.7: Sampling windows evaluated to find an optimal classification-reaction time ratio. The windows correspond to: 0.1s (cyan), 0.2s (red), 0.5s (green), 0.7s (purple) and 1s (orange). Recall that one second is our task limit time for achieving a suitable human-robot interaction.

the performance was the F1-Score, capturing both the precision and the recall of the test.

In order to fulfill the requirement of a useful robot reaction, the prediction time should be short enough so that the proposed methods apply to our realistic scenario. For that reason, it was not considered the whole sample's length but smaller portions (windows), which contained only the initial information. In total, five different window sizes were evaluated: 0.1, 0.2, 0.5, 0.7 and 1 second (see Fig. 3.7). The intuition is that the larger the sampling window is, the higher the chances to properly classify the human's intention will be, but the longer the operator would need to wait until the robot reacts to the interaction. Therefore, the aim is to find a trade-off between the prediction time and the classification performance. Our experience says that 1 second is a convenient amount of prediction time for an efficient and feasible human-robot collaboration. Thus, a longer inference time would be undesirable. Note that the total prediction time would include both, the sampling window's size, and the time the approach needs to infer the label of the sample.

3.5.2 Evaluation of the raw-data-based approach

The proposed method, 1NN+DTW, was evaluated, for each of the window's sizes previously defined, concerning the classification performance and the inference time per sample. Recall that two different implementations of multi-variate DTW were used, *dependent* and *independent*, from now on, DTWd and DTWi, respectively. Due to the *lazy learning* nature of the kNN classifier, it was also evaluated how the length of the samples fed to the classifier affected the inference time. In particular, the measurements of the windows were sub-sampled to smaller portions. Five different lengths were considered, which are expressed as the percentage of the window's length which remains after the sub-sampling: 100% (no sub-sampling), 8%, 6%, 4%, and 2%. Tables 3.1 and 3.2 show the results of the evaluation.

From Tables 3.1 and 3.2, we draw several conclusions. First, the bigger the window, the better the performance, see the evolution of F1-Score in Table 3.1. It is also true that the growth of the window's size results in an increment of the inference time per sample (see Table 3.2).

DTW method	F1-Score for different window's sizes				
	0.1 s	***0.2 s***	***0.5 s***	***0.7 s***	***1 s***
DTWd 100	0.889269	0.905781	0.975967	0.979997	0.987533
DTWi 100	0.906666	0.917357	0.978199	0.983574	0.988869
DTWd 8	0.885296	0.896193	0.968881	0.971941	0.991542
DTWi 8	0.901552	0.903042	0.965363	0.976463	0.991532
DTWd 6	0.881585	0.905163	0.9662	0.97196	0.992428
DTWi 6	0.89922	0.912131	0.967561	0.97512	0.991544
DTWd 4	0.870823	0.898531	0.963153	0.976416	0.985741
DTWi 4	0.888409	0.904715	0.964489	0.978246	0.985303
DTWd 2	0.868783	0.894189	0.951568	0.97818	0.985298
DTWi 2	0.8842	0.904574	0.954712	**0.979993**	0.98574

Table 3.1: F1-Score values for the different types of raw-data-based classification (*dependent* and *independent* DTW), sampling window's size (0.1, 0.2, 0.5, 0.7, and 1.0 secs.) and percentage of sub-sampling (where 100 means non-sub-sampling). The longer the sampling window was, the better the classification performance was. Observe how, for our task, a 0.5 seconds sampling window already provided a very good F1-Score. In bold, the value for the best possible combination considering the trade-off between recognition time and performance.

DTW method	Inference time (s) for different window's sizes				
	0.1 s	***0.2 s***	***0.5 s***	***0.7 s***	***1 s***
DTWd 100	1.27896	2.7022	6.5332	9.19757	12.6787
DTWi 100	1.39247	2.92458	7.26754	10.1909	14.111
DTWd 8	0.0433759	0.125803	0.426389	0.604001	0.895554
DTWi 8	0.045948	0.140452	0.469486	0.658161	0.989375
DTWd 6	0.0260689	0.0818415	0.279886	0.444692	0.700091
DTWi 6	0.0264917	0.0852878	0.2938	0.498601	0.735671
DTWd 4	0.0127826	0.043153	0.161143	0.267458	0.436421
DTWi 4	0.0138515	0.0450503	0.176923	0.28999	0.486605
DTWd 2	0.00802248	0.0124193	0.0548359	0.10122	0.169155
DTWi 2	0.0082175	0.0133079	0.0568831	**0.108486**	0.178205

Table 3.2: Inference time per sample for the different types of raw-data-based classification (*dependent* and *independent* DTW), sampling window's size (0.1, 0.2, 0.5, 0.7 and 1.0 secs.) and percentage of sub-sampling (where 100 means non-sub-sampling). The larger the window of data was, the longer the inference time was. The total time to recognize the operator's intent is the addition of the window's size plus the inference time per sample. In bold, the time for the best possible combination considering the trade-off between recognition time and performance.

This is reasonable since the kNN algorithm is a *lazy learner*. Any time a new sample is to be classified, the similarity between that sample and the rest of the training samples is computed. Hence, the longer the samples, the more time takes to compute the similarity, prolonging the whole inference process.

The aim here is to find the combination of DTW version, sampling window's size, and sub-sampling with the best compromise between F1-Score and total recognition time. DTWd with the window's size of one second and sub-sampling of 6% got the best F1-Score result (99.24%). The inference time per sample for this same case was above half a second (0.7 seconds). Hence, the total operator's intent inference time would be around 1.7 seconds, which is above the one second sought in this work, making it an invalid alternative. Indeed, any case which used the one-second window can be discarded for the same reason. Fortunately, reducing the window's size, while helping to reduce the inference time, did not decrease the performance too much. This was especially true for windows bigger than 0.5 seconds, which ensured values of F1-Score above 95%. The best F1-Score value for that window was around 97.8%, indeed a fantastic result. It corresponded to the case of not doing sub-sampling together with DTWi. Nevertheless, if we used that configuration for the approach, the time needed to infer the operator's intent would be above seven seconds, once again undesirable. Indeed, any case in which sub-sampling is not applied may be discarded since the inference time was always over the maximum desired time. Hence, the search was restricted to the cases with 0.5 and 0.7 second windows and sub-sampling, where the differences in F1-Score and time were marginal in most cases. It was selected a case in which the trade-off between inference time (0.8 seconds) and performance (97.99%) was rather good. This case corresponded to DTWi, a window of 0.7 seconds and sub-sampling of the data to the 2% of the window's size.

3.5.3 Evaluation of the feature-based approach

The proposed method, GPLVM+SVM, was evaluated for all the different already mentioned window sizes and with respect to both the classification performance and the inference time per sample. A priori, it was unknown which size of the latent space would produce a good performance. Therefore, different sizes of latent space were also evaluated: 2, 3, 5, 10, and 20 latent variables. Besides, the two types of GPLVM were evaluated too, optimized (GPLVM-op) and non-optimized (GPLVM), see Fig. 3.6 for more details. Tables 3.3 and 3.4 respectively summarize the results of the F1-Score and the inference time with respect to the different window's sizes and all the commented variations of GPLVM. Observe that in some cases where the window's size was very small (0.1 and 0.2 secs.). The shorter window outperformed by little the longer one. This behavior is counter-intuitive but possible due to the still negligible information contained within those small samples and the random selection of the training set.

From the results, it is interesting to note the effect of the optimization during the inference step in the GPLVM. The inference time per sample was always longer when GPLVM inference was optimized. Furthermore, that time grew accordingly to the number of latent variables (see

GPLVM method	F1-Score for different window's sizes				
	0.1 s	*0.2 s*	*0.5 s*	*0.7 s*	*1 s*
GPLVM-op 2	0.759941	0.743376	0.931787	0.953032	0.959269
GPLVM 2	0.761758	0.74465	0.930028	0.951698	0.958399
GPLVM-op 3	0.784054	0.813443	0.943897	0.96377	0.976454
GPLVM 3	0.780917	0.816794	0.942582	0.966424	0.979127
GPLVM-op 5	0.827579	0.84916	0.96592	0.972986	0.976502
GPLVM 5	0.809314	0.845254	0.968096	0.977827	0.984909
GPLVM-op 10	0.876251	0.871952	0.963224	0.975576	0.980847
GPLVM 10	0.838423	0.845608	0.965613	**0.982702**	0.991111
GPLVM-op 20	0.853017	0.875167	0.968448	0.980449	0.980437
GPLVM 20	0.850769	0.878743	0.968096	0.981384	0.993331

Table 3.3: F1-Score values for the different types of feature-based classification (*optimized* and *non-optimized* GPLVM inference), sampling window's size (0.1, 0.2, 0.5, 0.7, and 1.0 secs.) and number of latent variables (2, 3, 5, 10, and 20). Note that the bigger the number of latent variables was, the better the result was, which also happened with the window's size. In bold, the value for the best possible combination considering the trade-off between recognition time and performance.

GPLVM method	Inference time (s) for different window's sizes				
	0.1 s	*0.2 s*	*0.5 s*	*0.7 s*	*1 s*
GPLVM-op 2	0.644066	0.846901	0.651173	0.725712	0.723963
GPLVM 2	0.0939578	0.112943	0.0975801	0.134856	0.135385
GPLVM-op 3	0.671361	0.647414	0.694552	0.662102	0.833498
GPLVM 3	0.11745	0.0999844	0.110787	0.0909256	0.122548
GPLVM-op 5	0.809567	1.21396	0.945272	1.1361	1.20001
GPLVM 5	0.101104	0.0981438	0.113414	0.130257	0.132759
GPLVM-op 10	2.38703	1.86129	2.15111	2.4688	1.46768
GPLVM 10	0.116964	0.130362	0.115786	**0.119832**	0.134894
GPLVM-op 20	8.69852	7.2344	6.12122	5.11297	5.24385
GPLVM 20	0.107757	0.114087	0.114022	0.135751	0.138607

Table 3.4: Inference time per sample for the different types of feature-based classification (*optimized* and *non-optimized* GPLVM inference), sampling window's size (0.1, 0.2, 0.5, 0.7 and 1.0 secs.), and number of latent variables (2, 3, 5, 10 and 20). GPLVM-op led to longer inference time than GPLVM, which also applied when the number of latent variables grew. In bold, the time for the best possible combination considering the trade-off between recognition time and performance.

Table 3.4). Another interesting finding was that the inference time per sample, when there was no optimization, remained quite short and stable no matter the window's size nor the number of latent variables (see Table 3.4). Hence, in terms of inference time, GPLVM without optimization was preferred. Moreover, the difference in performance score between the

optimized and not optimized versions is negligible (see Table 3.3). This fact reinforces the previous finding, allowing us to conclude that the non-optimized version of GPLVM is the most convenient alternative.

Focusing on Table 3.3, it is observable that a higher number of latent variables produced a better F1-Score result. Specifically, for the cases of using two and three latent variables (especially two), the performance was usually much poorer. The best result in terms of performance, F1-Score of 99.33%, corresponded to the GPLVM version without optimization, the window of 1 second and 20 latent variables. The inference time per sample was around 0.14 seconds, so the total inference time was 1.14 seconds, slightly superior to the one second set as desirable. Hence, it seemed reasonable to reduce the window's size to 0.7 seconds. In that case, the best alternative was to use 10 latent variables, and again the non-optimized GPLVM. It would result in losing a bit of quality in the performance, from 99.33% to 98.27%, but decreasing the time from 1.14 to 0.82 seconds, fulfilling the recognition time requirements.

3.5.4 Comparison of raw-data-based and feature-based approaches

From the best combination of parameters for each of the two proposed methods, we aim to select one to be used during the qualitative study conducted with users presented in Section 3.6. The selected combination in the case of 1NN+DTW, ensured an inference time of 0.8 seconds and a performance score of 97.99%. It corresponded to use independent DTW, a window of 0.7 seconds and sub-sampling of the data to the 2% of the window's size (see Sec. 3.5.2 for more detail). When using GPLVM+SVM, the selection was GPLVM without optimization, a window of 0.7 seconds and 10 latent variables. This approach resulted in a F1-Score of 98.27% and an inference time of 0.82 seconds (see Sec. 3.5.3 for more detail). The quantitative differences between the two alternatives are marginal, which hinders the selection and makes necessary considering qualitative aspects. In robotics, especially in industrial environments, data is presented in heterogeneous ways: sequential data, digital, etc. In our use case, some examples may be variables encoding the previously inferred human intention, or whether the user is inside the workspace or not. Let's imagine that the previously inferred human's intention was *grabbing object*, using GPLVM+SVM, this could be added to the feature vector of latent variables and train the SVM classifier with the new extended vector. Therefore, it could be learned that when the robot does not hold the object, *polishing* cannot be performed. 1NN+DTW however, cannot deal with other data apart from sequential, thus it would be necessary to use a second kNN model with another metric (e.g. Euclidean) and then apply ensemble learning techniques. Based on this, we consider GPLVM+SVM to be a more versatile approach, thus it was used for the evaluation with users.

Used dataset	F1-Score for different window's sizes				
	0.1 s	*0.2 s*	*0.5 s*	*0.7 s*	*1 s*
Natural	0.838423	0.845608	0.965613	0.982702	0.991111
Mechanical	0.914686	0.943609	0.980069	0.987351	0.99131

Table 3.5: Evaluation over the *natural* and the *mechanical* data sets of the approach GPLVM+SVM without optimization and 10 latent variables. Using shorter window sizes (up to 0.5s), the results in the *natural* dataset are worse. For larger window sizes the model behaves similarly for both datasets.

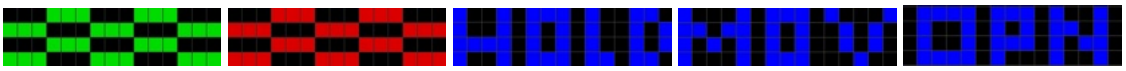
3.5.5 Comparison of natural and mechanical data sets

The performance of the selected method, GPLVM+SVM, is evaluated using each of the data sets. Recall that the chosen parameters are the non-optimized GPLVM inference with 10 latent variables. Although the selected sampling window's size was 0.7, the approach was tested for the usual five sizes used along the rest of the chapter. As before, cross-validation without replacement was used ten times, and the data was randomly split into training (75%) and test (25%).

Table 3.5 depicts the F1-Score values obtained from the evaluation of GPLVM+SVM against both data sets, demonstrating that the previous assumption was true. In general, using the *mechanical* dataset produced better results than utilizing the *natural* dataset. It is surprising that when the window's size was 0.2 seconds, the F1-Score was even close to 95%. However, for the window chosen for the validation with users, 0.7 seconds, the differences between the performance using any of the datasets are minimal. Therefore, the proposed approach works quite well even when the dataset contains more natural samples of physical human-robot interaction.

3.6 Validation - Recognizing operator's intent in a realistic scenario

To validate the selected approach, GPLVM+SVM, it was setup an experiment with several users individually collaborating with a robotic arm according to the industrial scenario of polishing car emblems. The validation was conducted using fifteen healthy individuals within the age range of 18 to 35. Users were selected among people who had knowledge about the robotics domain and had been in contact with robots before. People with reduced mobility or any cognitive disability which could affect the perception of the robot's behavior were not included. Each of the users received an individual explanation, no more than five minutes, about how they were expected to



(a) Ready. (b) Unconfident. (c) Hold. (d) Move. (e) Open.

Figure 3.8: LED patterns used by the robot to communicate with the user using the robot's armband. (a) Green pattern used to indicate when the robot is ready for physical interaction. (b) Red pattern indicating low classification confidence (<70%). Textual patterns showing the state of the robot when user intents are identified with high confidence: (c) 'hold' (polish intent), (d) 'move' (move intent), (e) 'open' (grab intent). The character 'e' could not be expressed due to the four-row armband matrix restriction.

interact with the robot. This included both general information about the system and particular notions about the expected movements for each of the three classes/intentions. Nevertheless, users were not allowed to train before the evaluation began, because then, it could be evaluated if there was an adaptation of the user to how the system inferred the different intentions. Users were also informed about their rights, possible risks, and they were asked to sign a consent form specifically designed for this experiment. The experiment protocol was favorably approved by the Human Subject Research Committee of the Spanish National Research Council (CSIC).

3.6.1 Validation setup

The proposed approach is validated in a setting inspired by a collaborative task in which the force exchange is not only present but fundamental for the accomplishment of the task. Using the force-based information, the robot should be able to identify the intent of the operator (Sec. 3.3.2) and to adapt its state/behavior to it. In order to ensure a mutual understanding between the human and the robot, the robot was equipped with a force sensor, used to measure the interaction from the human to the robot, and an armband made of LEDs through which the robot informed the user of its internal state. The latter displayed different patterns (see Fig. 3.8). During the validation experiment, the robot behaved according to the finite state machine captured by Algorithm 1.

Recall that the scenario was inspired by a real industrial case in which an operator is meant to inspect and polish car emblems. Please, refer to Figure 3.2a to see the different parts of the robot setup. We can only show the adapter where the emblem was attached to since emblems contain private commercial brand logos and can not be shown due to confidentiality agreements. During the experiment, users were in front of the robot so that the physical interaction was comfortable, and they had a rag used to polish. Figure 3.2b shows an example of the pose of a user while polishing, and a video of the validation with users is available online[4].

[4] www.iri.upc.edu/groups/perception/SIMBIOTS

Algorithm 1: Finite state machine for the control of the robot during the validation.

```
Data: Force sensor's signals
Result: Robot's state adaptation
while true do
    robot in initial pose;
    inform operator: the robot is ready for interaction;
    wait for physical contact;
    if detected physical contact then
        prepare sample from raw sensor data;
        infer operator's intention;
        if inference's confidence ≥ 0.7 then
            inform operator: next robot's state;
            adapt the robot's state to the inferred intention;
        else
            inform operator: the inference's confidence was low;
        end
    else
        do nothing;
    end
end
```

	Grab	***Move***	***Polish***
Grab	**0.6133**	0.3800	0.0067
Move	0.1200	**0.8667**	0.0133
Polish	0.0667	0.1667	**0.7667**

Table 3.6: Normalized confusion matrix of the performance of the system during the validation with all users and trials. Most of the samples of the classes 'grab' and 'polish' that are incorrectly classified are inferred as instances of 'move', indicating the existence of some bias in favor of the latter class.

3.6.2 Evaluation procedure

Each user (N=15) was asked to perform thirty trials randomly selected from the three operators' intent/actions explained in Section 3.3.2. Among the thirty trials, ten were forced to correspond to each of the three classes/intentions. Note that since trials were randomly arranged for each person, the possibility of finding bias in the evaluation due to the order of the trials was diminished. Both, the ground truth and the inferred value were annotated for each user's trial. For the evaluation, two different variables were studied: the overall performance of the system (confusion matrix), and the adaptation of the users throughout the experimental validation.

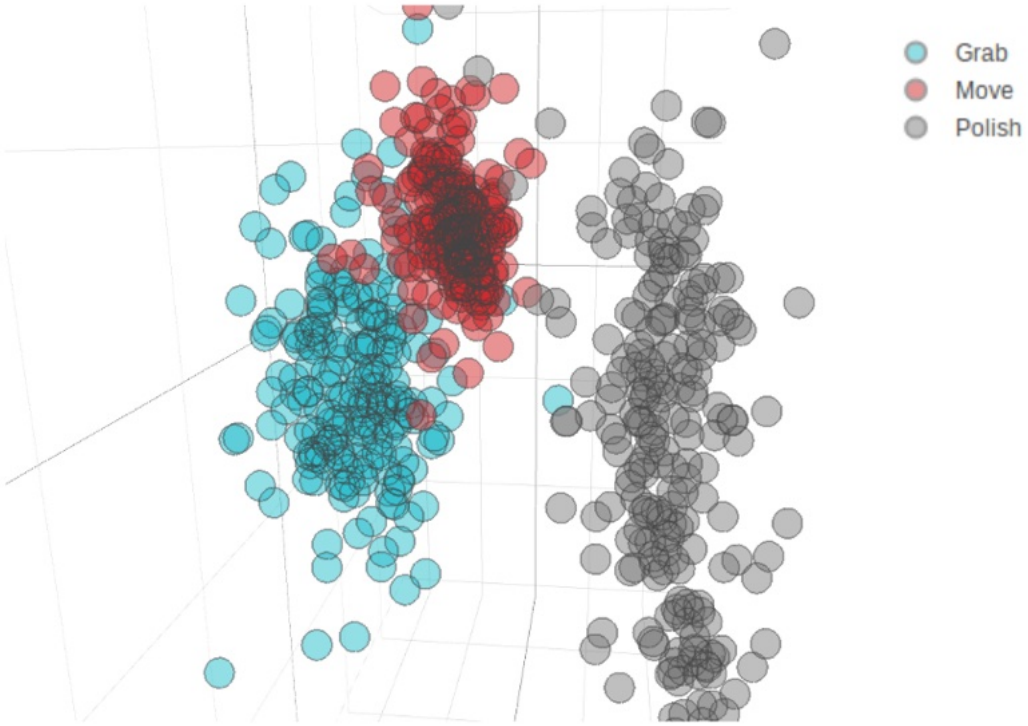


Figure 3.9: Single perspective of the data visualization using the three most significant latent variables from the original ten. The distribution of the data in this lower space shows that the samples of the class 'move' are rather close to the other two classes, which could be the cause of why the model seems to be a bit biased in favor of this class.

Table 3.6 contains the average confusion matrix for the systems' performance for all users. Note that the 'move' intent was the easiest to identify. Indeed, there was a large percentage of false positives for this class, a symptom of a clear bias of the model in its favor. This can be better understood by looking at the sample distribution in the three-dimensional space defined by the most significant latent variables among the ten used (see Figure 3.9). It can be observed how the samples from the 'move' class fall in the middle of the other two classes, which explains why there are many false positives. Indeed, the higher proximity between the 'move' and the 'grab' classes translates into a lager number of false positives.

It was also studied whether there was an adaptation of the users to the system, which would be observable in the performance of the system during the validation experiments. Recall that users only received a short explanation of the three classes and in which axes they could perform the movements for each action. There was ambiguity among classes and users had a particular way to move for each action. Hence, during the first trials, the system's performance was generally poorer. Here *adaptation* means that the users understood which

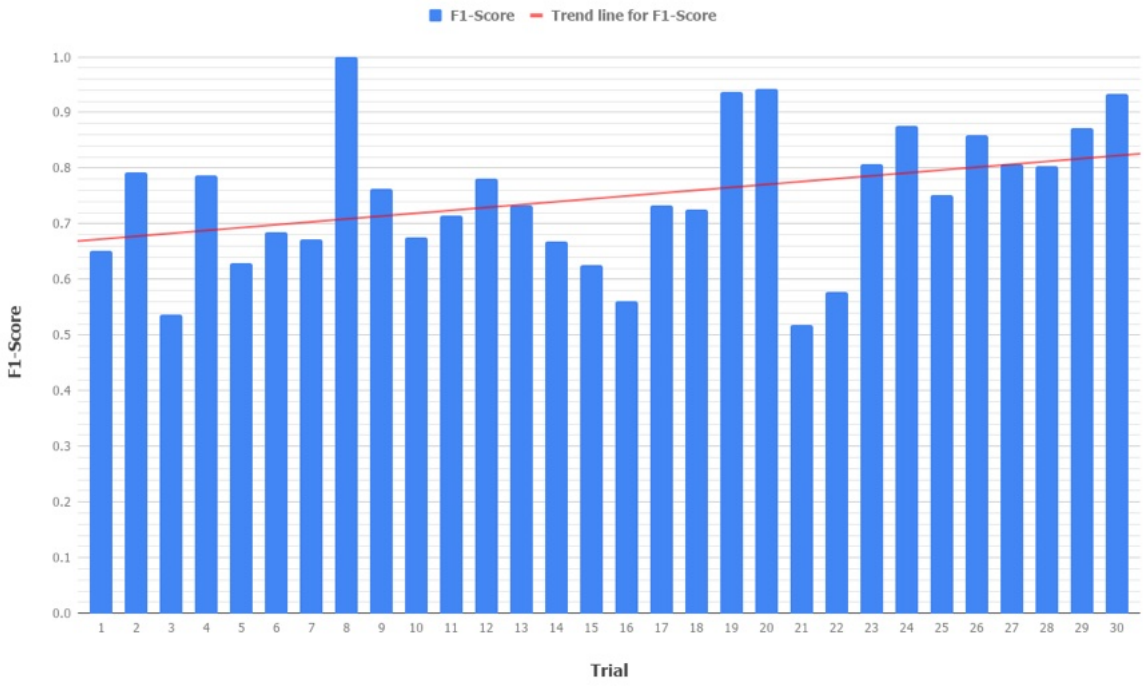


Figure 3.10: Averaged F1-Score of the system for all the users along with the experiment's trials. The positive slope of the trend line for the F1-Score is an indicator of the adaptation of the users to the system. Please recall that none of the users followed the same sequential trial set since they were randomly generated.

movements for each class ensured a better performance of the system. Note that this was possible because users could see the result of the inference. The averaged F1-Score was used to measure the performance of the system for all the trials and users, and the result showed a positive slope of the trend line for the F1-Score (Fig. 3.10). We consider that once the trend line was above 0.8, users would have already adapted. In our case, this corresponded to the last five trials of the experiment.

3.7 Discussion

In this chapter, we explored how to recognize and classify operators' intent while they interact with a robot in the execution of a collaborative task. The chapter presented three major contributions that are of specific relevance for perception tasks in collaborative robotics scenarios: a novel force-based dataset of physical human-robot interaction, force-based operator's intent inference approaches, and the validation with users of the whole system in a

scenario inspired by a realistic industrial application. In this work, the physical interaction between the robot and the human, not only existed but also played a major role since it was the main source of information for the robot to infer the human's intent. Were humans and robots to collaborate in the future, the main interaction would be physical.

Beyond these contributions, the hands-on experience gained from this chapter also inspired us to frame the scope of the conceptualized ontological models presented in this thesis. First, it became evident to us that the notion of 'collaboration' or 'collaborative event' should be conceptualized and modeled to ensure trustworthy human-robot collaboration. During the validation with users, it was interesting to see how confused they were when the robot inappropriately adapted after misclassifying their intention. This made us wonder questions such as: how a collaboration can be classified, or whether a collaboration can exist when the involved agents are not on the same page (i.e. they do not share a common intention or plan). These questions became essential when formalizing collaboration in Chapter 5. Second, this chapter also confirmed one of our initial intuitions. It would be interesting to explore and model the concept of 'adaptation', especially focusing on those situations in which a robot decides to adapt when collaborating with a human. Since robots will need to adapt and those adaptations might lead to human-robot misunderstandings, trustworthy robots should be able to reason about those adaptations (e.g. the motive to adapt). Chapter 5 introduces the formalization of this concept after having studied the different aspects of it. Finally, the validation with users made quite clear the need for explainable robots, and which might be some of their potential benefits. Using a simple LED armband already helped users to understand why the robot was not adapting accordingly to the interaction, which ensured mutual understanding. Of course, many other aspects of collaborative and adaptive experiences would require more expressive explanation modalities (e.g. text). Hence, Chapters 6 and 7 explore the use of textual explanations constructed from sound robots' ontological knowledge.

chapter four

Perceiving the risk of collision with humans in collaborative experiences

> *..the world is a dynamic mess of jiggling things, if you look at it right..*
>
> **— Richard Feynman**
> (Rubber Bands in Fun to imagine, BBC series)

Building on the findings of the literature review outlined in Chapter 2, this chapter continues the exploration initiated in Chapter 3. Hence, it delves into the knowledge to be conceptualized within robot recognition and decision-making tasks in human-robot collaborative scenarios. In those contexts, surely one of the most relevant aspects for robots to recognize is the potential risk of collision with their human collaborators, which would be used during robots' decision making. Indeed, since the robot did not move during the collaborative use case considered in Chapter 3, it makes sense to consider a different robotic scenario now. Hence, this chapter explores the concept of triggering safety stops in collaborative scenarios where humans and robots are in constant dynamic closeness. Time-to-contact (TTC) was used to recognize and categorize the risk of collision and to adapt the robot's behavior with respect to the detected risk. The work contributed with a novel formulation to compute TTC in a three-dimensional space, and an algorithm to stop the robot's motion based on the computed TTC. The approach was evaluated first, in simulation, and second, using a real robot and a simulated human (aiming for repeatability). In both, using prototypical motions to validate the

improvement of our approach in delaying the safety stop compared to two baseline methods. It was also conducted a qualitative validation through the implementation of our approach in a real use case: a complete collaborative task in which a human and a robot, closely interacting, filled a tray with tokens. Note that there is always a trade-off between safety and efficiency, and previous methods, while being protective, might tend to be quite simplistic obviating important dynamic aspects of the interaction that TTC captures. By the end of the chapter, it is also discussed how this work helped framing the scope of the conceptualized ontological models presented in this thesis. For instance, the research made clear the need for considering notions related to safety (e.g. risk) when conceptualizing the knowledge around collaborative and adaptive robot experiences. Indeed, a formal conceptualization of such notions would be of great help to regulate and certify autonomous robots to be used in human environments. Furthermore, the addition of explanations to inform users about the robot's estimation of risk, might also contribute to increasing the acceptance of robots.

4.1 Motive

In 2011, the International Organization for Standardization released the ISO 10218.1 [ISO 10218-1:2011, 2011] and the ISO 10218.2 [ISO 10218-2:2011, 2011], which presented safety guidelines for industrial robots. In 2016, the ISO/TS 15066 [ISO/TS 15066:2016, 2016] extended the previous standards providing specific guidance for safety in collaborative robotics, where a formulation to compute the minimum protective distance was proposed. One of the main limitations of this formulation is that the real direction of motion of the robot and the human is not taken into account. Hence, it results in an over-conservative risk estimator, and prevents a proper collaboration in applications where humans and robots constantly and closely share the workspace. Indeed, there is not a standard way to address this issue yet, and the actual implementation of the formula is still greatly left to the discretion of the integrator [Marvel and Norcross, 2017].

Inspired by the aforementioned ISO standards, several works about safety in collaboration have been published during the last years [Vogel and Elkmann, 2017, Nikolakis et al., 2019, Magrini et al., 2020, Zanchettin et al., 2015]. Indeed, some of them discussed and aimed to overcome different ISO formulation drawbacks [Vicentini et al., 2014, Campomaggiore et al., 2019]. However, there is still room for improvement, especially in collaborative tasks in which the human-robot closeness is regular (such as the one in Fig. 4.1). Hence, this chapter aims to explore an alternative to the ISO's formulation well adapted to intensive human-robot collaboration. Specifically, the concept of time-to-contact (TTC) [Hecht and Savelsbergh, 2004]

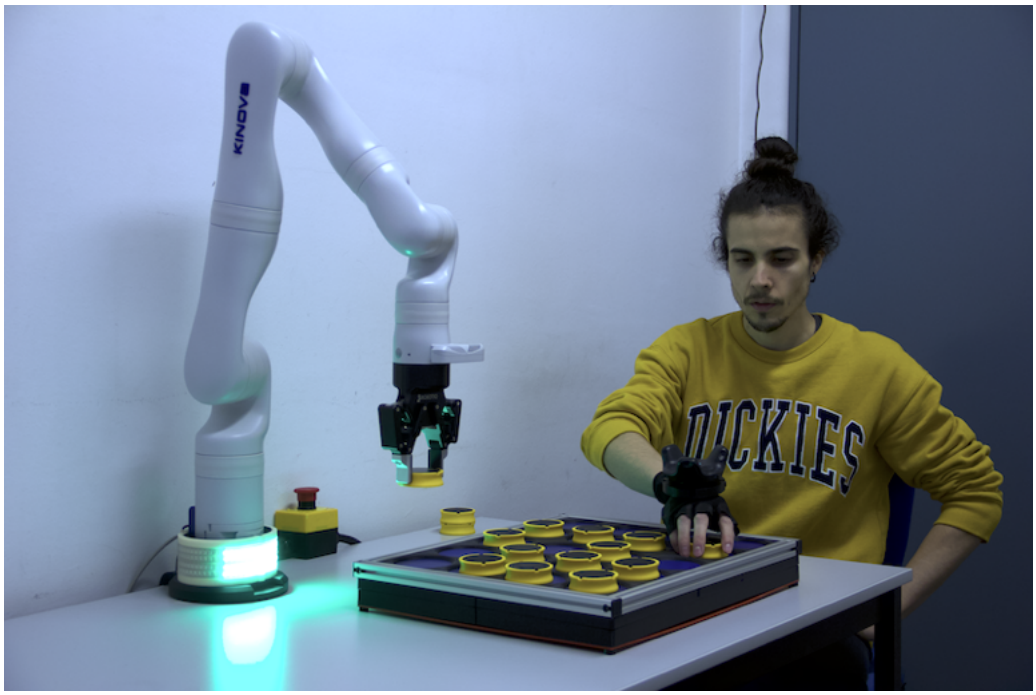


Figure 4.1: Collaboratively filling a tray: example of an industrial task where the human and the robot continuously share both the workspace and the execution of the task (pick and place).

is proposed as an indicator of the likelihood of the risk of collision during close human-robot collaborations. Unlike most of the approaches within the literature, TTC naturally captures the dynamics of the human-robot interaction, considering the actual pose, speed, and direction of both agents. This would make TTC a more precise risk indicator, delaying safety stops and allowing the robot to operate for longer times before stopping.

4.2 Related work

The ISO standards for safety in collaborative robotics [ISO 10218-1:2011, 2011, ISO 10218-2:2011, 2011, ISO/TS 15066:2016, 2016], proposed to use the Speed and Separation Monitoring (SSM) approach, maintaining a robot's speed and a minimum distance between the robot and the human. SSM is utterly aligned with the scope of this chapter, and it has been extensively applied in the collaborative robotics domain. For instance, some works proposed to adapt the robot's speed to the distance using discrete regions of the space [Vogel and Elkmann, 2017, Nikolakis et al., 2019, Magrini et al., 2020]. Aiming to enhance the collaboration's fluency, several authors suggested adapting the robot's speed with respect to the current human-robot distance continuously, but still without considering the human's and robot's

motion direction [Lasota et al., 2014, Zanchettin et al., 2015, Joseph et al., 2020, Rosenstrauch et al., 2018]. Looking for further improvement, more complex formulations to compute the human-robot distance considered the human's and/or the robot's motion direction [Vicentini et al., 2014, Byner et al., 2019, Campomaggiore et al., 2019]. In all those articles, as in this chapter, the robot's behavior is adapted based on the estimation of a possible risk of collision with a human. The estimation is based on the distance between the human and the robot, and the robot adapted its speed. This chapter proposes to simplify the formulation using the time-to-contact concept, as it naturally embeds the directions and velocities of the two agents.

Time-to-contact has been widely used in the literature for automotive collision estimation, warning, and avoidance [Li et al., 2016, Qu et al., 2018, Song et al., 2018, Yang et al., 2019]. Authors proposed different approaches: vehicle-to-vehicle and vehicle-to-driver warning, obstacle avoidance, autonomous emergency braking system, etc. In those works, TTC was computed in a two-dimensional space, since cars move in a plane. In this work, it is explored how the ideas discussed in those articles might be extrapolated to collaborative industrial scenarios, where humans and robots move in a three-dimensional space.

This chapter discusses an approach to stop the robot using the collision estimation metric, because that is the most compliant strategy with the regulations of industrial environments (e.g., ISO). However, TTC has also been used to control other robot's reactions such as adapting the speed or modifying the robot's motion plan for different applications: robot docking and landing [Kendoul, 2014, Zhang et al., 2017], unmanned aircraft system maneuvers [Eguíluz et al., 2020], and obstacle avoidance and target chasing [Kaneta et al., 2010]. Those works presented interesting approaches showing the potential use of TTC in different applications, which might be considered as inspiration for future work.

4.3 Time-to-contact-based safety stop for close human-robot collaborative experiences

4.3.1 Background on time-to-contact

TTC is a biologically inspired measure typically used for obstacle detection and reactive control of motion. It can be defined as the time that an observer will take to make contact with a surface assuming constant relative velocity. Hence, TTC is usually expressed in terms of the speed and the distance of the considered obstacle:

$$ttc = -\frac{Z}{\frac{dZ}{dt}}, \tag{4.1}$$

where Z is the distance between the observer and the obstacle, and dZ/dt is the velocity of the observer with respect to the obstacle. It is possible to compute TTC from a pure computer vision perspective, by just detecting the deformation of objects in consecutive RGB images without calibration [Alenyà et al., 2009, Kaneta et al., 2010, Garcia et al., 2016]. Those methods are very sensitive to error detection, so in this chapter TTC was computed utilizing the actual observer's and obstacle's poses and velocities, as there was access to these measurements. In this work, the observer will be the end effector of a robot and the obstacle the hand of a human.

4.3.2 Time-to-contact computation

Inspired by Hou et al. [Hou et al., 2014] and their TTC formulation for 2D collisions between circles, it is proposed here a 3D variation of their formula. Hence, the robot's end effector and the human's hand are considered as spheres, and it is computed the TTC as the time that it would take the two spheres to collide. Determining when two spheres collide is a matter of determining the moment at which the distance between their centers is equal to the sum of their radii.

Let $\vec{r'} = (r'_x, r'_y, r'_z)$ and $\vec{h'} = (h'_x, h'_y, h'_z)$ denote the positions of the robot and the human respectively at the moment of contact. Hence, the distance between them is

$$d = ||(r'_x, r'_y, r'_z) - (h'_x, h'_y, h'_z)||. \tag{4.2}$$

Knowing that when in contact the distance is equal to the sum of their radii and expanding Eq. 4.2 one gets

$$rad_r + rad_h = \sqrt{(r'_x - h'_x)^2 + (r'_y - h'_y)^2 + (r'_z - h'_z)^2}, \tag{4.3}$$

where rad_r and rad_h are the radii of the spheres representing the robot and the human respectively.

Assuming that the human and the robot are not currently colliding and that both move with constant linear velocity, their positions at the moment of contact can be rewritten based on their current position and velocity:

$$\vec{r'} = \vec{r} + \vec{v_r} ttc \tag{4.4}$$

$$\vec{h'} = \vec{h} + \vec{v_h} ttc, \tag{4.5}$$

where $\vec{r} = (r_x, r_y, r_z)$ and $\vec{h} = (h_x, h_y, h_z)$ denote the current positions of the robot and the human, $\vec{v_r} = (v_{r_x}, v_{r_y}, v_{r_z})$ and $\vec{v_h} = (v_{h_x}, v_{h_y}, v_{h_z})$ their current Cartesian velocity respectively,

Algorithm 2: decision-making loop to compute TTC, stop the robot and select the robot's speed

Input: Time-to-stop ($ttstop$), nominal robot's speed (V_{r_n}), robot and human radii (rad_r, rad_h)

1 $safety\_stop \leftarrow false$
2 **while *not*(*safety_stop*) do**
3 $info_r \leftarrow$ GetRobotPoseVelocity()
4 $info_h \leftarrow$ GetHumanPoseVelocity()
5 $ttc \leftarrow$ ComputeTTC($info_r$, $info_h$, rad_r, rad_h)
6 **if** *ttc* $\leq$ *ttstop* **then**
7 $V_r \leftarrow 0$
8 $safety\_stop \leftarrow true$
9 **else**
10 $V_r \leftarrow V_{r_n}$
11 **end**
12 Publish computed robot's desired speed V_r
13 **end**

and ttc is the corresponding time-to-contact. Substituting Eqs. 4.4 and 4.5 in Eq. 4.3, a quadratic equation is obtained where, if real positive roots exist, the smallest value is the pursued TTC.

4.3.3 TTC-based safety stop algorithm

Given a TTC value between the human and the robot, it can be used to adapt the robot's current state to avoid possible collisions. In this work, it is proposed to follow one of the strategies suggested in ISO 10218.1 [ISO 10218-1:2011, 2011]: issuing a safety-rated monitored stop. Hence, the robot would continue its motion until a certain TTC threshold is violated, from now on, time-to-stop (*ttstop*). Alg. 2 shows the decision-making process to compute TTC and stop the robot given that TTC value. If the value of TTC is equal or smaller than *ttstop*, the robot will stop its motion and a safety-rated monitored stop will be issued (see line 6 in Alg. 2). When the value of TTC is greater than *ttstop*, the robot will continue its motion at the task's nominal speed (see line 9 in Alg. 2). Once a safety stop was issued, the robot would remain stopped and it would exit the TTC-based safety stop loop (see line 2 in Alg. 2). Since the focus is on very close human-robot applications, the robot would only resume the motion after a human's command. This recovery strategy is the most appropriate one for industrial scenarios similar to the case discussed in this chapter.

4.4 Baseline approaches: ISO and Fuzzy ISO

In order to evaluate the proposed approach, two state-of-the-art metrics based on computing the minimum protective distance are used. First, it was used the linear version of the formulation defined in ISO/TS 15066 [ISO/TS 15066:2016, 2016]:

$$S = (v_h T_r + v_h T_s) + (v_r T_r + v_s T_s) + (C + Z_r + Z_s), \tag{4.6}$$

where S is the protective distance, v_h is the 'directed speed' of the operator (i.e., the rate of travel of the operator toward the robot), v_r is the directed speed of the robot in the direction of the operator, and v_s is the directed speed of the robot in the course of stopping. T_r is the time for the robot system to respond to the operator's presence, while T_s is the time to bring the robot to a safe, controlled stop. The remaining terms capture measurement uncertainty, where C is an intrusion distance safety margin based on the expected human reach, Z_r is the robot position uncertainty, and Z_s is the operator position (sensor) uncertainty. There is not a standard way to measure the human's and the robot's speeds, nor to compute the times. Indeed, setting the values of the uncertainty constants might also be challenging [Marvel and Norcross, 2017]. Since a discussion of the proper values to choose was out of the scope of this chapter, it was used a simplified version of the formula. First, it was assumed that the speed of the robot while stopping, v_s, was equal to the motion speed, v_r. Finally, it was obviated the effect of the uncertainty constants because they would equally affect all the methods compared in this work. The simplified equation is:

$$S = (v_h + v_r) T_b, \tag{4.7}$$

where T_b is the sum of T_r and T_s, thus the total time to brake, including the time to respond to the human's presence and to stop the robot. During the different evaluations, the value used for T_b was also used to set the time-to-stop ($ttstop$). Hence, establishing a correlation between our method (see Alg. 2) and the baselines presented in this section. Actually, it makes sense to use the total time to brake as the minimum TTC used to stop ($ttstop$). It would ensure that the robot would stop right before contacting the human, just as the minimum protective distance would.

The second state-of-the-art method used for the evaluations, Fuzzy ISO, was proposed by Campomaggiore et al. [Campomaggiore et al., 2019]. They presented a fuzzy-logic system to merge the protective distance formulation with information on the current human's and robot's motion direction. The fuzzy rules are used to scale the effect of the human's and the robot's

velocities: e.g. when they are going away in opposite directions their method allows them to relax the ISO's formula. The resulting formula would be the Eq. 4.7 multiplied by the output of the fuzzy-logic system $\alpha \in (0, 1)$:

$$S = \alpha(v_h + v_r)T_b. \tag{4.8}$$

Using the aforementioned two equations, the minimum protective distance (S) was computed. When the Euclidean 3D distance between the spheres representing the human and the robot was smaller or equal to S the robot would stop, analogously to the decision-making process proposed in Alg. 2.

4.5 Evaluating time-to-contact as the trigger of robot safety stop in close collaborative tasks

In order to evaluate the performance of the different safety stop methods, it was necessary to constrain all the possible situations that might occur in the target task (Fig. 4.1). After a comprehensive study, all possible robot-human collaborative dynamic states were summarized in a taxonomy of just 7 prototypical cases (see Fig. 4.2). For the evaluation, only the cases that would cause a safety stop were of interest, thus, the cases in which a collision might occur (see Fig. 4.2 a, c and e). From now on, they will be referred to as cases: A, C, and E respectively. It is important to remark that there is no need to evaluate the rest of the situations, nor the complete task execution. The reason is that there will be no change of behavior among methodologies.

4.5.1 Evaluation I - Statistical analysis in simulation

First, the performance of the proposed approach was evaluated against the two baseline methods presented in Sec. 4.4 (ISO and Fuzzy ISO). Recall that the evaluation was done considering only three cases from the taxonomy: A, C, and E.

For each case, two spheres moving close to each other were simulated. They represented the end effector of a robot and the hand of a human, from now on just robot and human. Once the safety stop was triggered, the simulation finished. The first hypothesis is that the proposed approach would allow the robot to move for more time before issuing a safety stop. The second hypothesis is that our approach would let the robot to get closer to the human but would still be safe, issuing a safety stop before any possible collision.

In order to validate the hypotheses, the approaches were evaluated for several human speeds in each of the three selected situations. Other parameters such as the human's and robot's initial

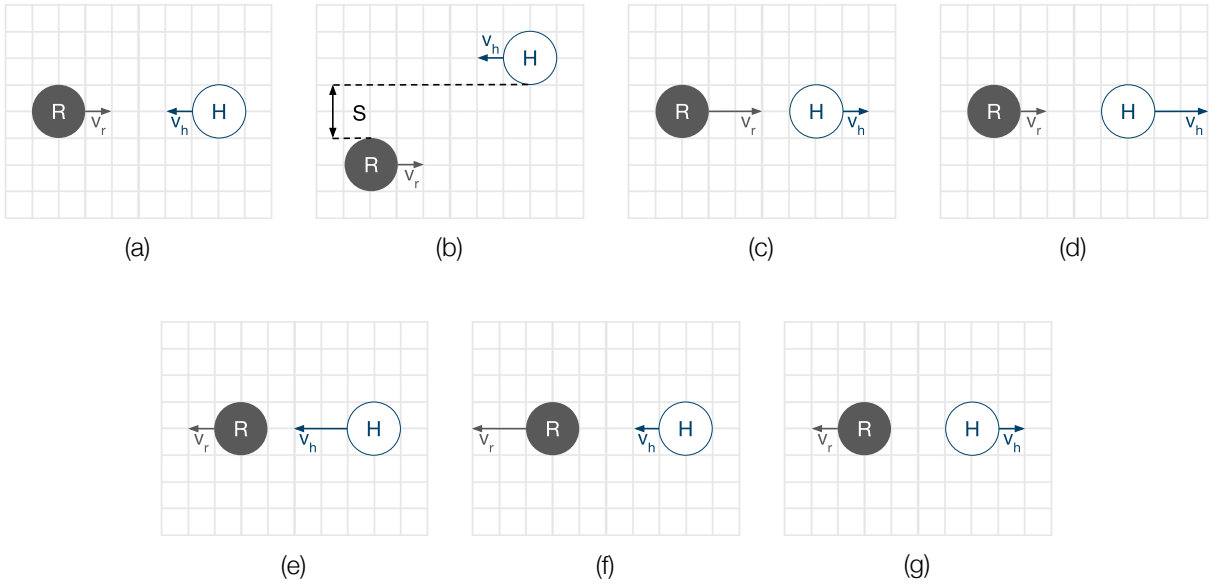
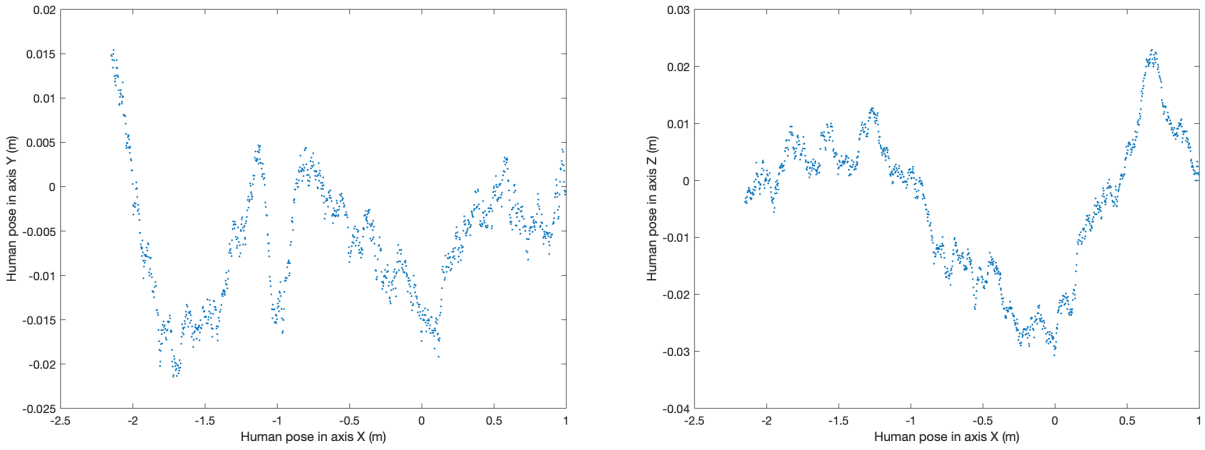


Figure 4.2: 2D symbolic representation of the prototypical human-robot collaborative situations. The module of the vector represents the robot's (R) and human's (H) velocity's magnitude (v_r and v_h). (a) Both agents approaching with probable contact. (b) Both agents approaching without probable contact. (c) Robot following the human with probable contact (the robot moves faster than the human). (d) Robot following the human without probable contact. (e) Human following the robot with probable contact (the human moves faster). (f) Human following the robot without probable contact. (g) Both agents getting away. Of all seven cases only three can trigger a safety stop due to a potential contact: (a), (c), and (e).

position or the robot's nominal speed, would affect the triggering of the safety stop. However, the focus was on the human velocity parameter for two reasons. First, it has a direct effect on the dynamics of the interaction, playing a fundamental role in the hypotheses. Second, it was a parameter that could not be controlled by the robot in real scenarios, thus it was worth studying how the three approaches reacted when it changed. Note that the human and the robot moved along one axis. A total of 1000 different human velocities were randomly generated, uniformly distributed within an interval of interest for each case. At each simulation timestamp, it was also added different white Gaussian noise to the three axes of each of the velocities.

The power of the noise for all three axes of motion was -20 dBW. The noise allowed us to simulate a realistic human motion, not just a straight line movement, as it can be seen in Fig. 4.3. The simulation time (10s) was selected to match the robot's speed (0.2m/s) and the maximum reach (1m) of the robot used in this work, Kinova Gen3. The frequency was 100Hz, and the $ttstop$ (see line 6 in Alg. 2) and the T_b (see Eq. 4.7) were both set to 0.5s. The radii of the spheres representing the human and the robot were fixed at 0.05m. The robot's nominal speed was 0.2m/s. The initial distance between the robot and the human was 0.9m, for case



(a) (b)

Figure 4.3: Example of the human pose evolution for a single simulated noisy motion. (a) and (b) show the evolution of the human pose along the planes XY and XZ, respectively. Note that a noisy human velocity was simulated at each simulation timestamp aiming for a realistic human motion, therefore, the pose evolution did not follow a straight trajectory.

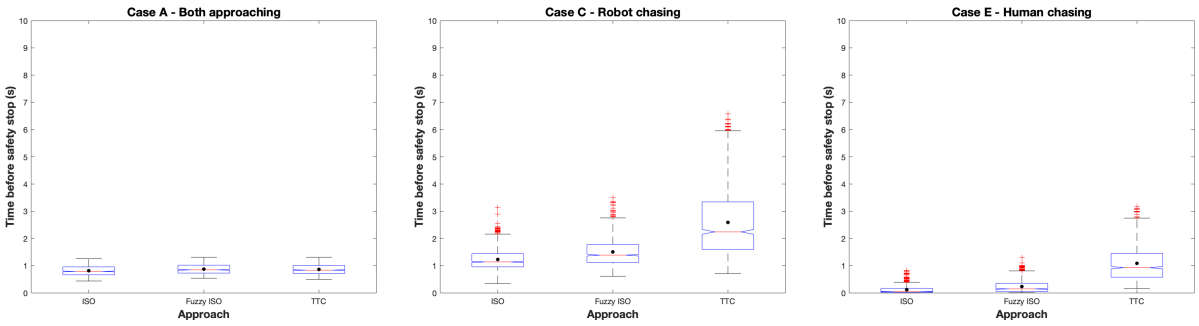
A, and 0.4m for cases C and E. The intervals for the randomly generated human speeds for the cases A, C and E were: $[0.3, 0.6]$, $[0.0, 0.15]$, and $[0.3, 0.6]$, respectively. Recall that the motion was mainly in one axis, although we added noise to it. Each case's main direction is depicted in Fig. 4.2.

Before a safety stop was issued, it was computed the time the robot was moving, and the final Euclidean distance between the human and the robot. The time before stopping allowed us to validate whether our method delayed the safety stop or not, confirming the first hypothesis. The distance was reported to show that our method allows the robot to get closer while still being safe (no collision), supporting the second hypothesis.

Evaluation I - First hypothesis: the robot moves more time

A statistical analysis was conducted to evaluate the significance of the proposed approach's improvement in delaying the safety stop with respect to the baseline methods: ISO and Fuzzy ISO. For that purpose, it was measured the time the robot moved before the safety stop was issued. Fig. 4.4 shows the distributions of measured time for each case and evaluated approach.

We manipulated the three different methods (independent variable) and assessed them with respect to the time before stopping (dependent variable), for each of the prototypical cases. First, Kruskal-Wallis was used to evaluate if there was a statistically significant difference in group mean, obtaining $\chi^2(2) = 55.08, p < 0.001$, for case A, $\chi^2(2) = 1113, p < 0.001$, case C, and $\chi^2(2) = 1812.8, p < 0.001$, case E. Second, as the results rejected the null hypothesis, it was



(a) (b) (c)

Figure 4.4: Distributions of the time the robot moved (a, b, c) for each simulated case and approach. The mean value is represented by a dot. TTC clearly outperformed the others in (b) and (c).

carried out a post-hoc analysis to find out where the differences occurred between the groups. A Dunn & Sidák post-hoc multiple comparison test revealed a significant pairwise difference between our method and the other two in two of the cases: C and E. In case C, the time before braking using TTC was significantly different from the ones using ISO and Fuzzy ISO with a p-value of 0 in both cases. In case E, TTC was also significantly different to ISO and Fuzzy ISO, with a p-value of 0 for both comparisons. These results proved that the differences in the values depicted in Fig. 4.4 are statistically significant. Hence, in these two cases, TTC allowed the robot to move for a longer time before stopping. This fact decreased the human-robot protective distance. Specifically, in case C, the robot using TTC moved a 110.52% more time on average than with ISO, and a 71.81% more than with Fuzzy ISO. In case E, the improvement was far greater, a 802.68% w.r.t. using ISO and a 358.83% w.r.t. Fuzzy ISO. These improvements would be notorious in long-term collaborations, especially in tasks that would imply medium and high levels of interaction (see Fig. 4.5). In case A, our method was significantly different from the ISO method with a p-value of $9.010e^{-8}$, while no significant differences were found between the Fuzzy ISO and TTC. However, the mean difference between TTC and ISO was really small (5.85%), as it is shown in the values depicted in Fig. 4.4. Note that in case A, the human and the robot approached each other, which is the default assumption that ISO's formulation does. Hence, it was expected that our method would capture the high risk of the situation and behave similarly to the ISO.

Evaluation I - Second hypothesis: the robot gets closer but it is still safe

A second statistical analysis evaluated the significance of our approach's improvement in allowing closer but still safe human-robot distances. The analysis was again performed with respect

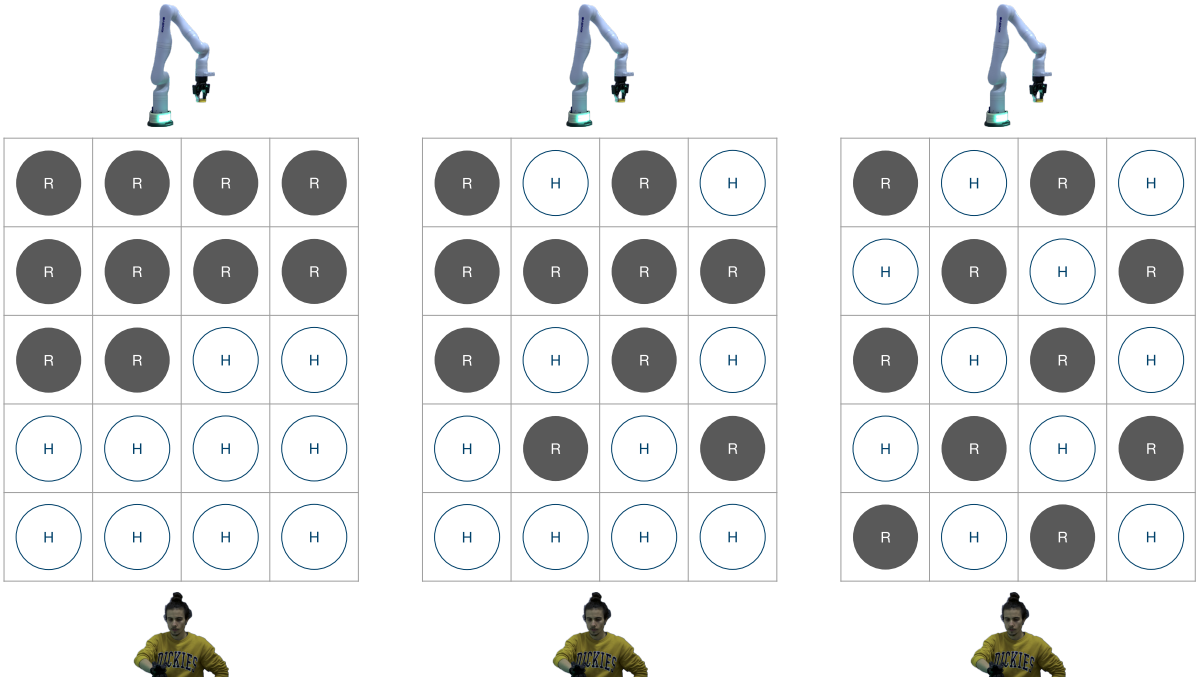
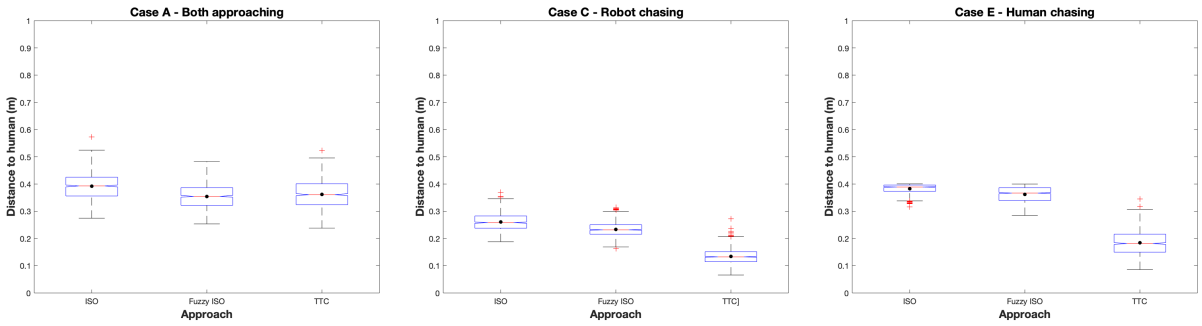


Figure 4.5: Examples of the task's distribution between the robot and the human with different levels of interaction. Grey circles (R) indicate the tray's compartments that the robot would fill, while white circles (H) would be filled by the human. (a) Low level of interaction with nearly zero possible crossing trajectories. (b) Medium level of interaction where a few of the robot's and human's targets might involve trajectories' intersections. (c) High level of interaction where the distribution of the task's targets would potentially cause crossing trajectories, leading to several probable contacts and safety stops.

to the ISO and the Fuzzy ISO methods. The final human-robot distance was measured once the safety stop was issued. Fig. 4.6 shows the distributions of measured distance for each case and evaluated approach.

We manipulated the three different methods (independent variable) and assessed them with respect to the human-robot distance after stopping (dependent variable). This is for each of the three prototypical cases. First, Kruskal-Wallis was used to evaluate if there was a statistically significant difference in group mean, obtaining $\chi^2(2) = 290.71, p < 0.001$, for case A, $\chi^2(2) = 2138, p < 0.001$, case C, and $\chi^2(2) = 2132.5, p < 0.001$, case E. Second, as the results rejected the null hypothesis, it was carried out a post-hoc analysis to find out where the differences occurred between the groups. A Dunn & Sidák post-hoc multiple comparison test revealed a significant pairwise difference between our method and the other two in the three prototypical cases: A, C and E. In case A, the distance after braking using TTC was significantly different from the ones



(a) (b) (c)

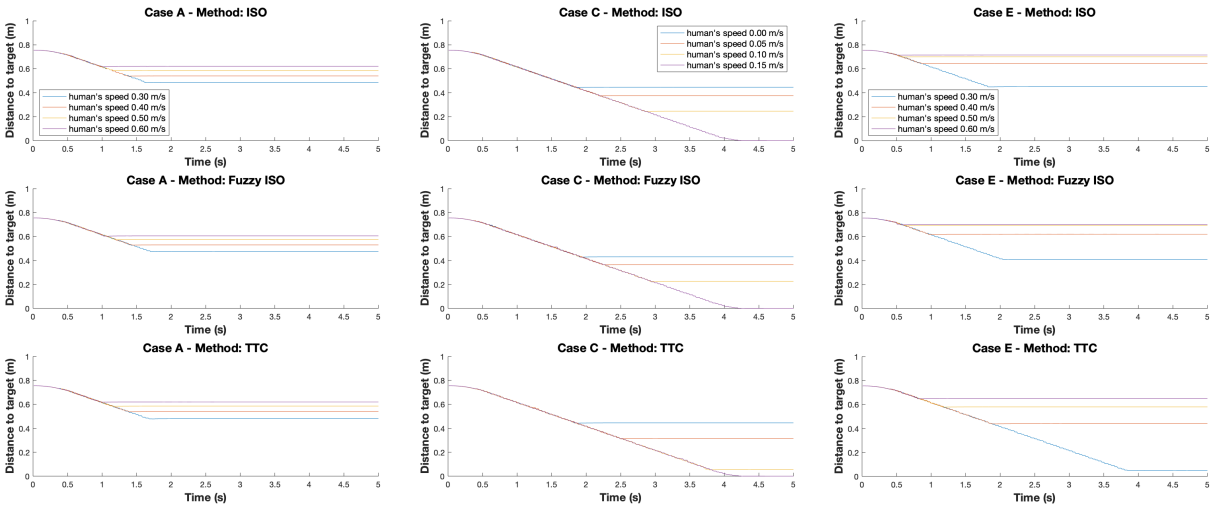
Figure 4.6: Distributions of the final distance to the human (a, b, c) for each simulated case and approach. The mean value is represented by a dot. TTC produces shorter distances in (b) and (c), but always ensuring safety and avoiding collisions.

using ISO and Fuzzy ISO with a p-value of 0 and 0.0018, respectively. In cases C and E, TTC was also significantly different to ISO and Fuzzy ISO, with a p-value of 0 for both comparisons.

These results proved that the differences in the values depicted in Fig. 4.6 are statistically significant. However, in case A, the mean difference in the human-robot distance produced by TTC was really small w.r.t. the other two methods. Specifically, using TTC the average final distance was a 7.69% smaller than with ISO and a 2.19% larger than with Fuzzy ISO. In cases C and E, the differences were larger and TTC allowed the robot to get closer to the human before stopping. Specifically, in case C, the robot using TTC produced a reduction in the final human-robot of a 48,51% and a 42.46% w.r.t. ISO and Fuzzy ISO respectively. In case E, the reduction was a bit larger, a 51.82% w.r.t. using ISO and a 49.06% w.r.t. Fuzzy ISO. Reducing the final human-robot distance might affect safety, but TTC produced no collisions during the simulation. Furthermore, the percentage reduction in the final distance is far smaller than the increase in the time the robot moves before stopping. Hence, we can say that TTC greatly improves productivity while slightly compromising safety.

4.5.2 Evaluation II - Real robot and simulated human (aiming for repeatability)

In this case, the previous simulation-based evaluation was contextualized, showing how the proposed method may be useful in the real task shown in Fig. 4.1. The objective was to evaluate our approach implemented in a real robot, avoiding the problem of repeatability of the human operator. Hence, a realistic setup was prepared, where a real robot moved towards a specific target pose. Meanwhile, the system was fed with the position and the velocity of a simulated human, which moved accordingly to the three cases evaluated before. Specifically, for the cases A and C, the real robot moved from the pose where the tokens would be grasped to the release



(a) (b) (c)

Figure 4.7: The plots show the evolution of the robot's distance to a target pose during the second evaluation: real robot and simulated human. (a) Both agents approach with probable contact. (b) Robot following the human with probable contact. (c) Human following the robot with probable contact. Once the safety stop was triggered, the human's simulation finished and the robot remained stopped. This is why the distance to the target becomes constant in the plots, implicitly representing the time the robot was moving before the safety stop.

pose of one of the tray's compartments. This would be the case of a robot picking a token and trying to place it on a compartment. For the remaining case, E, the real robot moved along the opposite trajectory. In this case, emulating when the robot would have already placed the token and it would go to pick a new one. In this evaluation, four human velocities were simulated, and it was measured the final Euclidean distance from the robot to the target pose after stopping. Note that the distance to the target is related to the two variables studied in Sec. 4.5.1: the robot's motion time and the distance to the human before stopping. Considering the previous evaluation's results, the hypothesis here was that the robot would clearly be able to get closer to the target pose using our method in cases C and E. The robot Kinova Gen3 was used equipped with the 2F-85 two-finger gripper from Robotiq, and the same parameters as in Sec. 4.5.1. The four different simulated human speeds were selected from the intervals used in Sec. 4.5.1 for the cases A, C and E: $[0.3, 0.6]$, $[0.0, 0.15]$, and $[0.3, 0.6]$, respectively. Some videos of this evaluation are shown in the additional material[1].

Fig. 4.7 depicts the evolution of the distance to the target for the four human speeds and each case. These results corroborated, in this case using a real robot, what was already obtained in simulation in Sec. 4.5.1. In case A, the differences between the three approaches

[1] www.iri.upc.edu/groups/perception/TTC

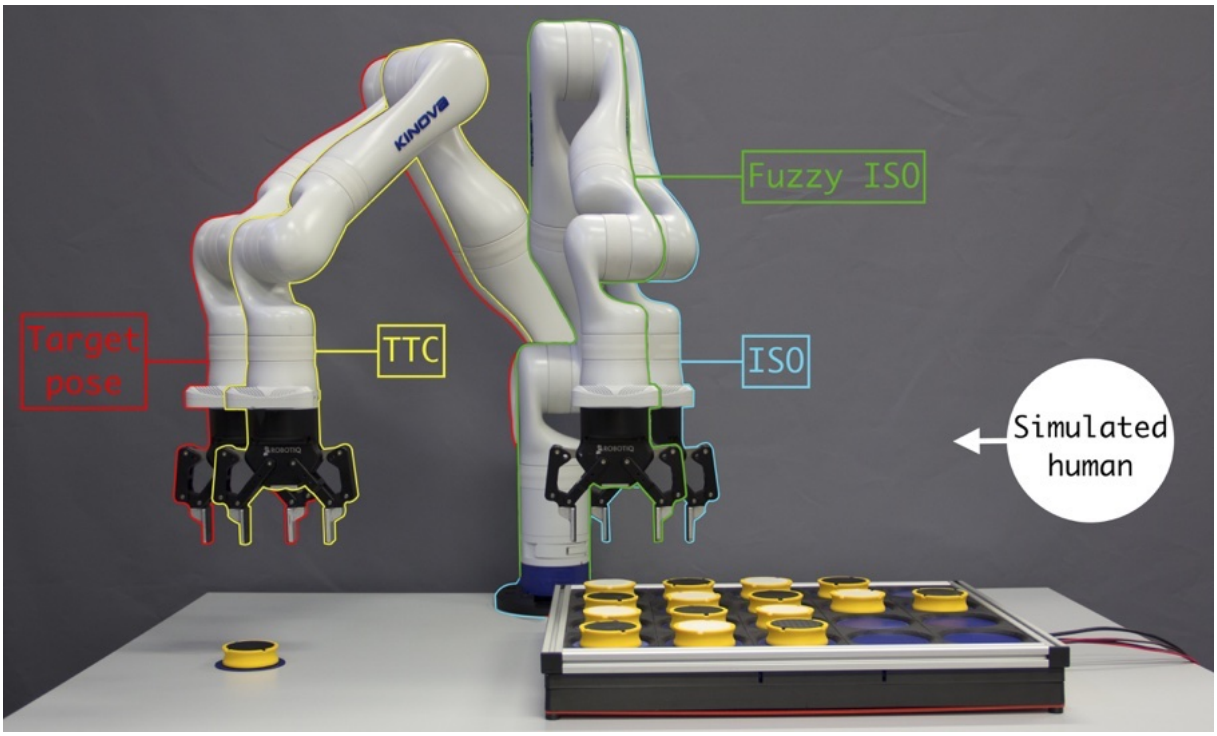


Figure 4.8: Exemplification of the final distance to the target pose during the second evaluation (case E): real robot and simulated human. The robot is trying to reach the target pose at 0.2m/s while the human is following the robot at a faster speed (0.3m/s in this case). In the image, we can see the target pose (red) and the final robot's pose after the safety stop issued by ISO (cyan), Fuzzy ISO (green) and TTC (yellow). As we can see, TTC allows the robot to get closer to the target before stopping.

remained minimal, around 2%. Cases C and E TTC again outperformed the other two approaches, allowing the robot to get closer to its target pose. Hence, our method let the robot get closer to finishing its task (e.g. placing a token) before stopping. Specifically, in case C, the final distance to the target using TTC was 23.22% and 19.92% shorter on average than with ISO, and Fuzzy ISO, respectively. In case E, the improvement is even greater, a 31.53% and a 29.16% shorter distance to target on average w.r.t. ISO and Fuzzy ISO, respectively. Fig. 4.8 depicts a comparison of the final robot's pose with respect to the target for the three methods in one of the experiments for case E.

4.5.3 Qualitative validation - Demo of a collaborative task with the real robot and a human

Finally, the proposed approach was implemented to be used in a realistic scenario where a robot and a human shared the task of filling the compartments of a tray. The same task will also be

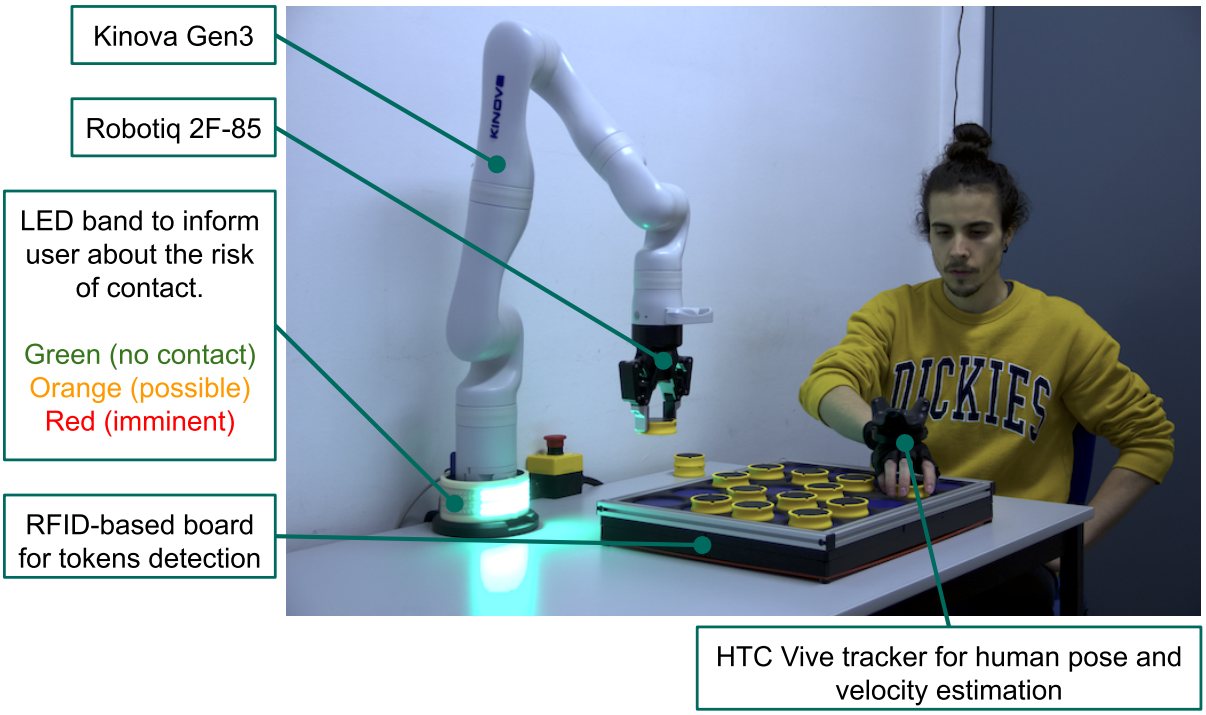


Figure 4.9: Setup for the demo of a collaborative task: filling a tray.

used for the evaluation of the conceptual modeling and ontology-based explanation generation discussed in Chapters 5 and 6, respectively. Hence, the symbolic model will be grounded directly from the task's data (Chapter 5), so that the explanations are built with both abstract knowledge and data (Chapter 6). The video of one of the task executions can be found in the additional material[2]. In this implementation, when the human filled one of the compartments the robot was meant to fill, the robot modified its plan and continued with the other free targets. Once a safety stop was issued, the robot was put in joint admittance mode (compliant), and it resumed the motion only after the human's command. The human's pose and velocity were measured using an HTC Vive tracker on their hand, and the tokens using an RFID-based board for fast and precise detection. The measurement rate of the HTC and the robot was 100Hz. The robot shared with the operator its interpretation of the collision's risk using the lights on the robot's base (see Fig. 4.9).

[2] www.iri.upc.edu/groups/perception/TTC

4.6 Discussion

In this chapter, it was studied how the concept of time-to-contact (TTC) can be of use for issuing a safety stop in close collaborative robotic scenarios. It was proposed a novel TTC formulation and an algorithm to activate a robot safety stop when there is a potential contact. The approach was evaluated against two state-of-the-art methods in a set of prototypical cases. First, in simulation, where a statistical analysis was performed to study the significance of the results. Second, with a real robot and a simulated human, aiming for a more realistic evaluation while ensuring human repeatability. In both evaluations and two out of three cases, our approach clearly produced a later safety stop than the other standard methods, increasing the time the robot is moving/working. In the remaining case, the differences between the three methods were too small to be relevant. Later stops resulted in shorter final distances between the human and the robot. However, the distance was always large enough to avoid collisions. Furthermore, the increment in the time the robot moves before stopping (productivity) is higher than the reduction in the final human-robot distance (safety). This work is a step forward to enabling robots to be closer to humans while sharing the execution of tasks safely with them.

Beyond those contributions, the hands-on experience gained from this chapter also provided intuition to frame the scope of the conceptualized ontological models presented in this thesis. First, it became evident to us that notions related to 'safety' or 'risk' should be conceptualized and modeled to ensure trustworthy human-robot collaboration. We think that a formal definition of those terms would facilitate the certification and regulation of the autonomous collaborative robots of the future, which will closely interact with humans. Note that regarding safety, the reaction time is a crucial element, thus, we think that the robot's decision-making process to stop should occur at the level of data. The process of abstracting the data into ontological entities and making the reasoning at a semantic level would probably require too much time, delaying the robot's reaction. However, the ontological conceptualization of safety-related terms would still play a relevant role in building trustworthy robots. Robots equipped with an ontological model of safety and risk would be able to reason about past risky experiences for introspection and learning. Furthermore, they could store the experiential knowledge for later use in the construction of monitoring reports or explanations. Indeed, the robotic task introduced in this chapter is later used to validate the conceptual modeling for collaborative robotics (including safety-related concepts), and the ontology-based explanation generation discussed in Chapters 5 and 6 respectively.

Part II

Ontological conceptualization and modeling for explainable robots

chapter five

Ontological modeling for robot reasoning in collaborative and adaptive experiences

> *..the more I think about language, the more it amazes me that people ever understand each other at all..*
>
> **— Kurt Gödel**

Chapter 2 revealed that the literature fell short of comprehensively addressing the use of ontologies to support the cognitive task of recognition and categorization. However, the inference power of ontologies would be a great tool to recognize and categorize different robots' experiences, thus this topic should be investigated. It was also discovered that focusing on modeling industrial applications could translate into the formalization of domain knowledge that is not covered in the literature (e.g. collaborative events, safety issues, etc.). During the evaluation with users conducted in Chapter 3, when the robot inappropriately adapted, users were confused. This raised questions such as whether a collaboration can exist when the involved agents are not on the same page (i.e. they do not share a common intention or plan), or how to model the cases in which robots adapt their plans to the changes in the environment (e.g. the motive to adapt). Hence, it was concluded that the notions of 'collaboration' and 'adaptation' should be conceptualized and modeled to ensure trustworthy human-robot collaboration. Furthermore, modeling those concepts would certainly support the foundations of ontology-based explainable robots. The introduction in Chapter 1 discusses the idea that explainable agency requires functional abilities such as: reporting the actions they

executed (e.g. collaboration with humans), and explaining how actual events diverged from what was planned and how agents adapted to it (i.e. adaptation). Finally, Chapter 4 emphasized the importance of considering an ontological conceptualization of safety-related terms, which would play a relevant role in building trustworthy robots. Robots equipped with an ontological model of safety and risk would be able to reason about past risky experiences for introspection and construction of monitoring reports or explanations.

This chapter addresses the challenges raised during the previous chapters of the thesis by introducing the Ontology for Collaborative Robotics and Adaptation (OCRA). An ontology especially designed to represent, recognize and categorize the relevant knowledge entities in *collaborative scenarios* where robots *adapt* their plans to the ongoing changes in the environment. The use of the ontology is validated in a realistic case study in which a human and a robot share the execution of a task. First, it is shown the capability of the ontology to answer a set of competency questions in a contextualized scenario. Second, it is discussed how the formalization would work in some limit cases in which wrong instances of *collaborative* and *adaptive* events were purposely defined. OCRA is the very first ontology that allows to formalize and reason about the execution of human-robot collaborative tasks, robot plan adaptation, and different types of collaboration types and risks. Furthermore, it models knowledge that is relevant to the development of some of the main functionalities of explainable robots (e.g. explaining executed tasks and changes during the plan execution).

5.1 Motive

During the last decade, the industrial sector has shown a growing interest in more flexible manufacturing processes where humans and robots are expected to work together. For that purpose, collaborative robots, or co-bots, are robots specifically designed for direct interaction with humans within a collaborative workspace [ISO 10218-2:2011, 2011]. Implementing industrial processes where robots and humans collaborate, opens several questions such as how to cope with uncertainty and safety. Hence, collaborative robots shall be able to, among others, reason about their tasks' requirements (e.g. safety, performance, etc.), about the changes in their environment, and about the plan adaptations due to those changes.

The use of industrial collaborative robots has drawn the attention of many researchers, becoming a prolific research domain [Gervasi et al., 2020, Gualtieri et al., 2021, Kim et al., 2021]. Indeed, several works have discussed safety in collaborative robotic scenarios [Vicentini, 2020, Gopinath et al., 2021, Liu and Wang, 2021]. Furthermore, due to the usual high-productivity requirements of manufacturing processes, some authors have

(a) (b) (c)

Figure 5.1: Examples of collaborative tasks in which the human and the robot continuously share both the workspace and the execution of the task. (a) asking the robot for a tool, (b) collaboratively fill a tray with tokens (this task was used during the validation), (c) hand-over of a tool in which the robot and the human exchange forces.

researched the trade-off between productivity and safety [Zanchettin et al., 2019, Scimmi et al., 2021]. Meanwhile, others have proposed adaptive robotic solutions for industrial applications [Levine and Williams, 2014, Levine and Williams, 2018, Villani et al., 2019]. This large list of promising works has also come with some drawbacks. The lack of consensus on the meaning of concepts such as collaboration and adaptation has hindered the coherent development of methodologies and techniques. This has already shown to be a problem in safety applications, where the use of this terminology to assess risks might lead to confusion and potentially mistaken implementations [Vicentini, 2020].

A common approach to harmonize terminology and enhance its reusability is to use knowledge representation formalisms such as ontologies. Indeed, the use of ontologies has spread in the industrial domain, where the modular and reusable nature of this formalism has been of great help [Borgo et al., 2019a, Karray et al., 2019, Mohd Ali et al., 2019]. The 1872–2015 IEEE Standard Ontologies for Robotics and Automation [Schlenoff et al., 2012] presented a core ontology for robotics and automation, which is currently being extended to other robotics' sub-domains [Fiorini et al., 2017]. Furthermore, ontologies have been widely used for autonomous robotics during the last years [Olivares-Alarcos et al., 2019a], and one can even find some initial steps towards ontologies for collaborative robotics [Umbrico et al., 2020]. However, in none of these works can one find a comprehensive analysis and formalization of the notions this chapter focuses on: 'collaboration' and 'adaptation'. Hence, this chapter discusses a novel ontological conceptualization of those notions and other terms that are related (OCRA), which can be useful in different kinds of collaborative tasks (see Fig.5.1).

5.2 Related work

The 1872–2015 IEEE Standard Ontologies for Robotics and Automation [Schlenoff et al., 2012] was conceived as a reference for knowledge representation and reasoning in the domain, and a formal vocabulary for humans and robots to share knowledge about robotics and automation. However, it did not cover terminology for particular robotic sub-domains. Hence, several ontology-based systems for autonomous robots were implemented focusing on more specific notions. Some examples are Knowrob [Tenorth and Beetz, 2009, Beetz et al., 2018], ORO [Lemaignan et al., 2010], PMK [Diab et al., 2019] and CARESSES [Bruno et al., 2019a]. These works have explored and proven the relevance and usefulness of ontologies in robotics. However, they did not address the terminology defined in OCRA.

Some other authors have focused on industrial robotic applications. Stenmark et al. [Stenmark and Malec, 2015], proposed the ROSETTA ontology, aimed at supporting reconfiguration and adaptation of robot-based manufacturing cells. Balakirsky [Balakirsky, 2015] implemented an ontology-based system for automatic recognition and adaptation to changes in manufacturing workflows. Stipancic et al. [Stipancic et al., 2016], proposed to use a set of ontologies to semantically enrich the robot sensors data in order to enhance the decision-making process in a multi-agent scenario. Chen et al. [Chen et al., 2021], presented an ontology for automatic disassembly applications to represent terms related to processes, tools and production pieces such as fasteners. Although relevant for their domains, none of these works provided a formal definition for the concepts discussed in this chapter. Of special interest is the work of Umbrico et al. [Umbrico et al., 2020], who defined an ontology for human-robot collaboration. They focused on terminology which was mostly different to the notions defined in our work. Indeed, both ontologies could coexist and complement each other. The only overlap was regarding the notion of `Collaboration`. The definition proposed in this chapter is stronger and more general because it is based on a thorough analysis of how the concept was defined in the literature. Hence, it does not only represent a single perspective but also a view shared by several works, including theirs. Furthermore, their ontology lacked other notions covered in OCRA such as `Collaboration Place`, or `Plan Adaptation`.

Finally, one can also find several works about ontologies for the industrial domain in general [Liang, 2018, Sampath Kumar et al., 2019, Karray et al., 2019, Smith et al., 2019, Borgo et al., 2019a, Mohd Ali et al., 2019, Liang, 2020]. Nonetheless, the content defined in OCRA cannot be found in any of them.

5.3 OCRA - Ontology for Collaborative Robotics and Adaptation

There are several methodologies to help the knowledge engineer in the ontology construction process, e.g., [Fernández-López et al., 1997, Spyns et al., 2008]. Due to the variety of possible cases and the needed characteristics of the ontologies, none emerged as a definite standard. Furthermore, those methods are not suitable for developing an ontology from a foundational viewpoint where the characterization of the core concepts is more important than the coverage of the application domain, as in this case. Thus, this work relies on ontological analysis, an approach that precedes the usual ontology construction process and aims to fix the core framework for the domain ontology. This choice led us to perform the following steps: to set the ontology domain and scope (competency questions), to reconsider other conceptualizations (selection of relevant literature), to enumerate, analyze and compare existing concepts (identification of shortcomings), to develop and formalize a more solid conceptualization, and to create instances of the concepts and show their use (implementation/validation). Of course, there is some circularity in the actual procedure since this is a process of conceptual discovery and (re-)organization. As a final step, it is also considered the documentation and maintenance of the proposal. Note that to take the most out of different ontological languages, in this chapter the whole ontology is first formalized using FOL. In this way, the obtained model can express exactly what the notions mean. An OWL 2 DL version of the ontology is also provided, which contains less knowledge but can be used for computational purposes and, therefore, implemented in a real robot for run-time reasoning.

5.3.1 Scope, goal and competency questions

In order to develop OCRA, we followed a top-down approach. Hence, our ontology was built upon other higher-level ontologies. Specifically, we developed an ontology that is compliant with Knowrob [Tenorth and Beetz, 2009, Beetz et al., 2018], the most widely used knowledge-based framework for robots. Therefore, we inherited the use of its upper ontology, the DOLCE+DnS Ultralite (DUL) foundational ontology[Borgo et al., 2021]. Nevertheless, the concepts presented in this work are general enough to be adapted to and used with other upper ontologies. Furthermore, note that Knowrob is a system that has consistently improved over the last decade. This justifies using DUL and helps to generalize our work, since we could take advantage of some of their framework tools and experience.

OCRA was designed to represent relevant knowledge in the collaborative robotics domain, with a special focus on collaboration and robot plan adaptation. A group of questions is proposed

and a set of requirements on the content, which scope and delimit the subject domain that has to be represented in the ontology. Particularly, OCRA should be able to answer the following questions:

- Ontology coverage questions:

  C1 - What is a collaboration?

  C2 - What is a plan adaptation?

- Competency questions:

  Q1 - Which and how many collaborations are running now?

  Q2 - Which is the plan of a collaboration?

  Q3 - Which is the goal of a collaborative plan?

  Q4 - Are these agents collaborating?

  Q5 - Where is a collaboration happening?

  Q6 - How is a collaboration classified (e.g. non-physical)?

  Q7 - Which is the risk of a collaboration?

  Q8 - Which and how many plan adaptations are running now?

  Q9 - Which is/are the agent/s participating in the plan adaptation?

  Q10 - Why is an adaptation of an agent's plan happening?

  Q11 - Which is the plan before and after an adaptation?

  Q12 - Which is the goal of the agent involved in the adaptation that is also the goal to be achieved by both the old and the new plan?

5.3.2 On the meaning of Collaboration

Rationale - Ambiguity in the literature

The Oxford Dictionary defines `Collaboration` as *'the act of working with another person or group of people to create or produce something'* [OED, 2024]. This informal definition would let us talk about collaborative events. However, a formal definition is needed to enable robots to reason about these events. In this section, several informal definitions from the literature are analyzed, highlighting their differences and common points, and motivating the need for a comprehensive formal model for `Collaboration`.

In 2011, the International Organization for Standardization released the ISO 10218.1 [ISO 10218-1:2011, 2011] and the ISO 10218.2 [ISO 10218-2:2011, 2011], which defined

`Collaboration` as *'a special kind of operation between a person and a robot sharing a common workspace'*. Vicentini [Vicentini, 2020], discussed the ambiguity in the collaborative robotics' terminology. He stated that at least, *'there is a predominant consensus in assigning the concept collaboration to continuous, purposeful interaction associated with potential or accidental physical events (contacts)'*. The Organization for Economic Co-operation and Development, defined the collaborative problem-solving competency as: *'the capacity to engage in a process whereby two or more agents attempt to solve a problem by sharing the understanding and effort required to come to a solution'* [OECD, 2017]. Oliveira et al. [Oliveira et al., 2007], defined collaboration session (CS) as *'an event that is composed of the actions of its participants. A CS has one or more objectives, defining its main purpose'*. Dillenbourg [Dillenbourg, 1999], discussed the definition of collaborative learning. He stated that *'collaborative situations involve symmetry between what agents know and do, shared goals, and a low division of labor'*. Silverman [Silverman, 1992] defined `Collaboration` as *'the mutual sharing of goals in completing the tasks'*. Terveen [Terveen, 1995], defined `Collaboration` as *'a process in which two or more agents work together to achieve shared goals'*. He also derived a set of fundamental issues from his definition: agreement on the goal, plan and coordination, shared context and understanding of the current situation, communication, and adaptation and learning. Kolfshoten [Kolfschoten, 2007] studied several definitions of `Collaboration` and proposed a refined one: *'a joint effort toward a goal. This implies that all participants make an effort, combine it and direct it to achieve a desired state or outcome (goal)'*. Bauer et al. [Bauer et al., 2008], surveyed the human-robot collaborative domain, for them, collaboration means *'working with someone on something, aiming at reaching a common goal. To work cooperatively on something the partners need to agree on a common goal and a joint intention (plan) to reach that goal'*. Ajoudani et al. [Ajoudani et al., 2018], reviewed the state-of-the-art on human-robot collaboration. They considered that human–robot collaboration *'falls within the general scope of human–robot interaction, and it is defined when human(s), robot(s) and the environment come to contact with each other and form a tightly coupled dynamic system to accomplish a task'*. Note that the interaction or contact might also be non-physical (e.g. mental). Umbrico et al. [Umbrico et al., 2020], defined the concept collaborative process as *'a process, in order to represent production events that modify over time the state of the production environment from an initial situation to a final/resulting one'*. Their formal definition was the closest one to ours, although it lacked explicit mention of the shared plan and goal, only focusing on how a collaboration changes the environment. Hence, we think that our definition is more general, and theirs might be considered as a specialization of ours.

Even though all these definitions diverge, it is possible to find some patterns that most of them follow: collaborative agents must share a goal and a plan (understanding/coordination),

Definition source	Formal	Goal	Plan	Interaction/Execution
[ISO 10218-2:2011, 2011]	No	-	-	Yes
[Vicentini, 2020]	No	Yes	-	Yes
[OECD, 2017]	No	Yes	Yes	Yes
[Oliveira et al., 2007]	Yes	Yes	-	Yes
[Dillenbourg, 1999]	No	Yes	Yes	Yes
[Silverman, 1992]	No	Yes	-	-
[Terveen, 1995]	No	Yes	Yes	Yes
[Kolfschoten, 2007]	No	Yes	Yes	Yes
[Bauer et al., 2008]	No	Yes	Yes*	-
[Ajoudani et al., 2018]	No	Yes*	-	Yes
[Umbrico et al., 2020]	Yes	Yes*	Yes*	-
Ours	Yes	Yes	Yes	Yes

Table 5.1: Set of main aspects related to 'Collaboration' extracted from the literature. 'Formal' shows whether the literature definition was formalized or not. 'Goal', 'Plan' and 'Interaction/Execution' columns indicate whether the notion of each aspect was captured or not by the definition. (*Implicit in the definition).

and there must be interaction [Borgo, 2019] between them while executing the plan. Table 5.1 depicts a summary with these main aspects of `Collaboration` for each of the studied articles.

Definition in natural language

Considering all the aforementioned definitions, `Collaboration` is usually defined as a special kind of spatio-temporal entity (an event). Furthermore, it is often related to a goal and a plan, and it requires interaction among the agents. Based on this, the proposed novel definition of `Collaboration` is:

Definition 5.1. *Collaboration is an event in which two or more agents share a goal and a plan to achieve the goal, and execute the plan while interacting.*

Interaction is used as an unspecified term as it belongs to a higher level ontology, it is thus considered a primitive concept in OCRA. Informally speaking, interaction is *'the act of communicating with somebody, or having an effect on each other'* [OED, 2024]. For example, during a collaboration, a robot and a human interact when they exchange forces, and also when the robot is sharing its perception of the safety situation (e.g. by voice or lights). Note that our definition states that the collaborative agents shall share a plan and the goal to be achieved. Hence, even when an agent delegates a part of a plan, one shall understand that the agent maintains co-responsibility for that part of the plan. For example, let's consider that there is a robot and a human that are collaborating to fill the different compartments of a tray with work pieces, thus a collaboration exists as long as:

- the robot and the human share a Plan to fill the tray. Note that the plan can include generic activities, like 'picking some pieces and placing them on the tray until it is full', or more specific activities like 'picking the pieces starting from the closest and placing them on the tray from left to right';
- the robot and the human share the Goal to be achieved by the execution of the shared plan. It may be general, e.g. 'all tray's compartments with a piece', or specific, e.g. 'each tray's compartment filled with a certain piece'; and
- the robot and the human execute the shared plan while they interact, thus, they share an understanding of who is in charge of what during the execution.

Formalization in FOL

Formalizing the proposed definition of `Collaboration`, many classes and relationships were reused from the foundational ontology DUL (note the prefix *'dul.'*). The final formalization in FOL is:

$$\begin{aligned} Collaboration(e) \equiv dul.Event(e) \wedge \\ \exists y, z, p, g, t\,(y \neq z) \wedge dul.Agent(y) \wedge dul.hasParticipant(e, y) \wedge \\ dul.Agent(z) \wedge dul.hasParticipant(e, z) \wedge dul.Plan(p) \wedge dul.Goal(g) \wedge \\ dul.hasComponent(p, g) \wedge executesPlan(e, p) \wedge dul.hasTimeInterval(e, t) \wedge \\ \forall x\,(dul.Agent(x) \wedge dul.hasParticipant(e, x)) \rightarrow hasPlan(x, p, t) \wedge hasGoal(x, g, t). \end{aligned} \tag{5.1}$$

The definition reads as follows: *a collaboration is an event (e) in which at least two agents (y and z) participate, it is the execution of a plan (p) with some goal (g), and for any agent (x) in the collaboration its aim is to execute that plan and to achieve that goal.*

Note that the definition was not restricted stating that the pursued goal must be achieved at the end of the collaboration, thus, being general and considering cases in which the goal of the collaboration is not achieved (and perhaps, unknown to the agents, even not achievable). Furthermore, the relationship 'executes plan' was used here as a primitive which means 'following the sequence of actions in the plan'. Hence, we did not consider this notion in the strictest sense, which would be to execute the whole plan. This predicate holds between an event and a plan that is executed by that event. It was also found necessary the use of two new relationships that were not explicitly defined in DUL: 'has plan' and 'has goal'. They relate an agent with a plan and a goal, respectively, during a time interval. First, 'has plan' means that 'an agent intends to execute a sequence of actions (plan)'. Second, 'has goal' implies that 'an agent desires to achieve a goal'.

5.3.3 On the meaning of Adaptation

Rationale - Ambiguity in the literature

The Oxford Dictionary [OED, 2024] defines `Adaptation` as *'the action or process of changing something, or of being changed, to suit a new purpose or situation'*. This informal definition would be helpful to talk about adaptation events. However, a formal definition is needed to allow robots to reason about these events. In this section, several informal definitions from the literature are analyzed, spotlighting their discrepancies and shared points, and encouraging the need for a comprehensive formal model for `Adaptation`.

Järvenpää et al. [Järvenpää et al., 2016], presented an adaptation approach for small-size production systems, in which `Adaptation` *'referred to all controlled changes the production system goes through during its life cycle'*. Martín et al. [Martín H. et al., 2008], proposed a mathematical model of the phenomenon of `Adaptation`. Specifically, they defined a Law of Adaptation: *'every adaptive system converges to a state in which all kind of stimulation ceases'*. For them, an adaptive system *'has at least one process which controls the system's adaptation to increase its efficiency to achieve its goals'*. Lints [Lints, 2012], identified and discussed the main aspects of adaptation from different fields of research. He defined `Adaptation` as *'a process to change something (itself, others, the environment) so that it would be more suitable or fit for some purpose than it would have been otherwise'*. Smit and Wandel [Smit and Wandel, 2006], reviewed the concept of adaptation regarding humans' adaptation to global changes such as climate change. The authors stated that `Adaptation` *'might refer to a process, action or outcome in a system, in order for the system to better cope with, manage or adjust to some changing condition, stress, hazard, risk or opportunity'*. Smit et al. [Smit et al., 2000], discussed that a thorough description of adaptation should specify the system of interest that adapts, the stimulus that causes the adaptation, and the involved processes and their outcomes. Gjorven et al. [Gjorven et al., 2006], considered `Adaptation` as a service, and defined it as *'a service whose input event is an adaptation trigger, and whose output events are a set of services that potentially has been modified or produced during the adaptation'*.

All these definitions are ambiguous, but there are some patterns that most of them follow: adaptation shall be triggered by a stimulus, shall occur on an entity that would change to a new state, and shall aim to continuously pursue the achievement of a goal. Table 5.2 depicts a summary with these main aspects of `Adaptation` for each of the definitions.

Definition source	Formal	Trigger	Entity	Change	Goal
[Järvenpää et al., 2016]	No	-	Yes	Yes	-
[Martín H. et al., 2008]	Yes**	Yes	Yes	Yes	Yes
[Lints, 2012]	No	-	Yes	Yes	Yes
[Smit and Wandel, 2006]	No	Yes	Yes	Yes*	Yes*
[Smit et al., 2000]	No	Yes	Yes	Yes*	-
[Gjorven et al., 2006]	No	Yes	Yes	Yes*	-
Ours	Yes	Yes	Yes	Yes	Yes

Table 5.2: Set of main aspects related to 'Adaptation' extracted from the literature. 'Formal' column shows whether the literature definition was formalized or not. 'Trigger', 'Entity', 'Change' and 'Goal' columns indicate whether the notion of each aspect was captured or not by the definition. *Implicit in the definition. **Mathematical model but not an ontological one.

Definition in natural language

After studying the state-of-the-art, we thought that providing a general definition of `Adaptation` would be extremely challenging. Barandiaran et al. [Barandiaran et al., 2009], discussed that adaptation involves a norm specifying which is the appropriate change to make. Hence, depending on the type of norm, we could find different types of adaptations: task or plan-based, evolutionary, ecological, etc. In this work, we focused on plan-based adaptations, changes aimed at continuously pursuing the completion of a goal given an unexpected state or situation. Hence, we proposed the following definition of `Plan Adaptation`:

Definition 5.2. *'Plan Adaptation is an event in which one (or more) agent, due to its evaluation of the current or expected future state, changes its current plan while executing it, into a new plan, in order to continuously pursue the achievement of the plan's goal.'*

From the definition, one can extract the conclusion that if a plan was changed before starting its execution, that would not be an adaptation. Also note that if a change was part of a plan, we would not consider it to be an adaptation. Hence, if a robot's plan included two optional executions, choosing one would not be an adaptation. Indeed, some authors claimed that the capacity to adapt depends on the observer who chooses the scale and granularity of description [Di Paolo, 2005, Barandiaran and Moreno, 2008]. For instance, in a micro-scale, obstacle avoidance might be seen as an adaptive behavior, but in an environment rich in obstacles, it would not. For instance, let's consider the previous example where a robot and a human collaborate to fill the different compartments of a tray with work pieces, thus a plan adaptation exists as long as:

- the robot has a plan, and it executes it while the perception of a current or future state (situation) triggers the adaptation. A possible plan could be 'moving to a compartment to

release a piece', and the trigger might be 'the compartment is full'; and

- the robot changes its plan by no longer executing the action required by the previous plan, and from now on executes the new plan. Still aiming to fill the tray, the new plan could be 'moving to another free compartment'.

Formalization in FOL

In order to formalize the natural language definition of `Plan Adaptation`, again it was reused as much content as possible from the foundational ontology DUL (note the prefix *'dul.'*). The final formalization in FOL is:

$$\begin{aligned} PlanAdaptation(e) \equiv dul.Event(e) \wedge \\ \exists s, g, a, o, n, i, f, p, q\, dul.Situation(s) \wedge dul.Goal(g) \wedge dul.Agent(a) \wedge \\ dul.hasParticipant(e, a) \wedge dul.Plan(o) \wedge dul.hasComponent(o, g) \wedge \\ dul.Plan(n) \wedge dul.hasComponent(n, g) \wedge dul.hasPostcondition(i, s) \wedge betterPlan(s, n, o) \wedge \\ dul.Event(i) \wedge dul.hasTimeInterval(i, p) \wedge dul.Event(f) \wedge dul.hasTimeInterval(f, q) \wedge \\ p < q \wedge i + f = e \wedge executesPlan(i, o) \wedge executesPlan(f, n) \wedge \neg executesPlan(f, o) \wedge \\ \forall x\, ((dul.Agent(x) \wedge dul.hasParticipant(i, x)) \rightarrow hasPlan(x, o, p) \wedge hasGoal(x, g, p)) \wedge \\ \forall x\, ((dul.Agent(x) \wedge dul.hasParticipant(f, x)) \rightarrow hasPlan(x, n, q) \wedge hasGoal(x, g, q)). \end{aligned} \tag{5.2}$$

The definition reads as follows: *a plan adaptation is an event (e), with at least one agent (a), which is the change of a plan (o) with a goal (g) into a new plan (n) with the same goal, where the change is due to the evaluation that the situation s holding after the first part of the event (i) makes plan n in the second part (f) better than continuing plan o, and in the first part any agent (x) aims to execute plan o, while in the second part any agent aims to execute the plan n, and every agent has always the same goal (g).*

Note that at least one agent participates in the whole adaptation event while other agents may change due to the adaptation. A new predicate/relationship was included, 'better plan', which relates two plans and a situation that makes one of the plans better to achieve a goal. Hence, one could use this relation to state that a situation has caused one plan to be no longer good, and a new plan is better for accomplishing a goal. Note that one could similarly define 'worse plan' as its inverse predicate if required.

5.3.4 Complementary terminology

Rationale - Common terms in the literature

In the collaborative robotics literature, apart from the concept of `Collaboration`, it is widely spread the use of terms such as workspace, safety, or collaboration types [Vicentini, 2020, ISO 10218-1:2011, 2011, ISO 10218-2:2011, 2011]. Most of this terminology is already defined in well-established ISO standards, so one could just reuse the notions. However, there are no formal standard definitions yet, so this chapter formally defines the concepts as part of OCRA. Hence, allows robots to reason about the place where they collaborate, safety aspects, and the different types of collaboration.

ISO 10218.1 [ISO 10218-1:2011, 2011] defined `Collaborative Workspace` as *'a workspace within the safeguarded space where the robot and a human can perform tasks simultaneously during production operation'*. This definition is broad enough to capture most of the collaborative scenarios found in the industry, where a fixed workspace is often designed for collaborations. However, aiming to be general, this concept shall also be considered from the perspective of the place/environment where a collaboration occurs. In that case, the place could dynamically change due to the collaboration needs (e.g. if the collaborators have to do operations using machines in different areas of the shop floor). Indeed, some authors defined a collaborative dynamic geometrical region that includes the intersection of both the robot's and the human's workspace [Melchiorre et al., 2021]. Hence, two different concepts are defined: one for the notion of the place where a collaboration occurs (`Collaboration Place`), and another one for the common industrial notion of a fixed place for collaborations (`Collaborative Place`).

Regarding safety, the standard is to follow the guidelines of the ISO 12100 [ISO 12100:2010, 2010], which focuses on machinery's risk assessment and risk reduction. ISO 12100 defined risk as *'combination of the probability of occurrence of harm and the severity of that harm'*. Recall that the notion of risk has already appeared in this thesis when investigating time-to-contact as a collision risk indicator (see Chapter 4). In this work, the content from the ISO is combined with our previous experience with risk indicators to formalize and define the concept `Collaboration Risk` (see Sec. 5.3.4 for more details).

Finally, it would be interesting to classify different types of collaboration, so that robots could behave differently depending on each type. The ISO 10218.2 [ISO 10218-2:2011, 2011] defined four different collaborative operational modes for robots. They are useful for talking about different robot behaviors or strategies, but they cannot directly be considered as sub-classes of collaboration. Bauer et al. [Bauer et al., 2016], proposed a classification of different collaboration levels: cell, coexistence, synchronized cooperation, and collaboration. However,

these categories are ambiguously used in the literature [Vicentini, 2020], and they might lead to confusion. For the time being, there is not a standard taxonomy of collaboration types, actually, there can be many depending on the application domain. Hence, this chapter focused on a classification that is relevant for the target application and for risk analysis, which is based on the degree of physical human-robot interaction: `Non-physical Collaboration`, `Indirectly Physical Collaboration`, and `Directly Physical Collaboration`. In the future, other classifications might be considered to extend OCRA.

Definition and formalization

In OCRA, a `Collaboration Place` *'is the spatial location or the place of a collaboration'*. This concept was formally defined as a sub-class of `Place` (DUL) that `is location of` a `Collaboration`. Note that this definition focuses on the existence of a collaboration and where it is located. Hence, a collaboration place is the union of the spatial locations of all the entities involved in the collaboration, which could change over time. For instance, if the agents involved in the collaboration move to other places, the collaboration place would also move.

It was also defined `Collaborative Place` as *'a role of a place that is specifically dedicated to collaborations'*. It was formalized as a sub-class of `Role` (DUL) that `classifies` a `Place`. Recall that this definition focuses on the place where collaborations can occur. It is meant to capture the traditional view of an industrial collaborative workspace, in which a collaboration is only considered inside of a fixed workspace. It is worth noting that when a collaboration occurs in a place whose role is to be a `Collaborative Place`, there is also a `Collaboration Place` that can be different from the first one. For instance, when a collaboration is occurring at a work cell that plays the role of a `Collaborative Place`, if one of the agents goes out of the work cell to do part of the collaboration, the place where the collaboration happens (the `Collaboration Place`) would be different to the work cell.

Concerning safety, and based on the ISO 12100 [ISO 12100:2010, 2010], `Collaboration Risk` was defined as *'a quality that has a value used to characterize a collaboration, or a part of it, which combines the probability of occurrence of a given harm and the severity of that harm during that collaboration'*. It was formalized as a sub-class of `Quality` (DUL) that `is quality of` a `Collaboration`.

Regarding the different types of collaboration, it was first defined `Non-physical Collaboration` as *'an event type that classifies a collaboration, or a part of it, in which the involved agents do not exercise any physical force'*. For instance, selecting the next part of the plan to execute, asking for a tool (see Fig. 5.1a), verbally communicating commands or recommendations to collaborators [Nikolaidis et al., 2018, Chacón et al., 2020], or monitoring

how a part of a plan is executed by another agent. Second, `Indirectly Physical Collaboration` is *'an event type that classifies a collaboration, or a part of it, in which the involved agents exercise physical forces but they do not physically restrict the freedom of movement of any of the other agents'*. For instance, when a robot moves close to a moving human without exchanging forces (see Fig. 5.1b). Third, `Directly Physical Collaboration` is *'an event type that classifies a collaboration, or a part of it, in which the involved agents exercise physical forces, and they do physically restrict to some degree the freedom of movement of at least one of the agents'*. This includes the cases where the involved agents exchange contact forces, directly or through an object, as shown in Fig. 5.1c. Hence, any movement of one of the agents would affect some other agent. For instance, a collaborative hand-over [Pan et al., 2019], the collaborative task of polishing an object [Olivares-Alarcos et al., 2019c], or the assembly of a piece of furniture [Rozo et al., 2013].

In the case in which the freedom of movement is restricted by rules such as those related to safety (e.g. robot stops if a human is closer than a given distance), a collaboration would still be considered as `Indirectly Physical Collaboration` because the movement of the robot is restricted by the safety behavior, not by a pure physical impediment. Of special interest would be the case when a robot and a human [Pan et al., 2019], or two collaborative robots [Garcia-Camacho et al., 2020], are holding a deformable object. If both agents held it close enough so that there was still freedom of movement, we would consider it as a `Indirectly Physical Collaboration`. If they went further, so that the deformable object is completely stretched/extended, then we would be in a `Directly Physical Collaboration`, because if one of the agents moved some of the others would be affected by it. This example is related to a collaborative hand-over, a task that might also be categorized under other characteristics (e.g. robots' adaptability/responsiveness). In the future, other features and tasks will be considered to enlarge the list of collaboration types included in OCRA.

The three concepts were defined as sub-classes of `Event Type` (DUL) that `classify` a `Collaboration`. It was considered the option of defining them as sub-classes of `Collaboration`. Nevertheless, although they were useful concepts for reasoning, the differences between them were not ontologically meaningful for us. Furthermore, note that it was intentionally avoided the use of the concept 'contact' in the definitions. The proposed classification is more general since it also considers other physical forces, not only whether or not the human and the robot are touching each other.

5.3.5 OCRA formalization in OWL

Complementing the formalization in FOL, it is also provided an OWL 2 DL version of the ontology. It contains less knowledge but can be used for computational purposes and can be implemented in the robot for run-time reasoning. The ontology was implemented using Protégé [Gennari et al., 2003], and the developed OWL file is publicly available together with other additional material[1] to facilitate reuse and comparison.

Most of the axioms that were defined in FOL were translated into OWL 2 DL, with the exception of the three ternary relationships: `has plan`, `has goal`, and `better plan`. FOL supports the use of ternary relationships, but OWL 2 DL does not (although in some cases one can overcome this problem [Rector and Noy, 2006]). First, `has plan` and `has goal` were defined as ternary to express that agents had a goal or a plan during an interval of time. However, in the OWL 2 DL version of OCRA, the two properties only relate agents with their plans and goals, without stating for how long those relationships hold. This is not necessarily critical since the use of OWL 2 DL at run-time happens while the agents do have the plan and goal. For a broader use of the OWL 2 DL formalization, this issue can be solved by 'reification', introducing several relations $hasGoal\_tp(r, g)$, one for each instant (tp) in which the relation holds. For instance, if the plan is to move objects and we do that at a frequency of 1 per minute for a total of 1 hour, one could imagine checking every minute whether the agents maintain the plan. For this, it is enough to introduce 3600 relationships $hasGoal\_tn(r, g)$ with n going from 0 (initial time) to 3599. This solution is activity-dependent so it is not presented in the general definition. Furthermore, one can also exploit a temporal history of the knowledge base's facts (episodic memories) [Beetz et al., 2018]. Hence, one could determine the temporal interval during which a relationship holds (e.g. the time an agent has a goal or a plan). Second, `better plan` was defined as ternary to model that, given a situation, a plan is better than another plan. In the OWL 2 DL version of OCRA, there is one relationship that substitutes the ternary one: `is better plan than`, relating two plans. Since the relationship `is better plan than` is evaluated at the time when the situation s holds, the notion formalized in OWL 2 DL is a good approximation of the original one in FOL. Finally, note that other complementary relationships were included (e.g. the inverse of all the previous ones).

5.4 Validation I - Answering the competency questions

In this section, the use of OCRA is qualitatively validated in a lab mock-up of a real task, where a robot and a human share the task of filling the compartments of a tray (see Fig. 5.2). The video

[1] www.iri.upc.edu/groups/perception/OCRA

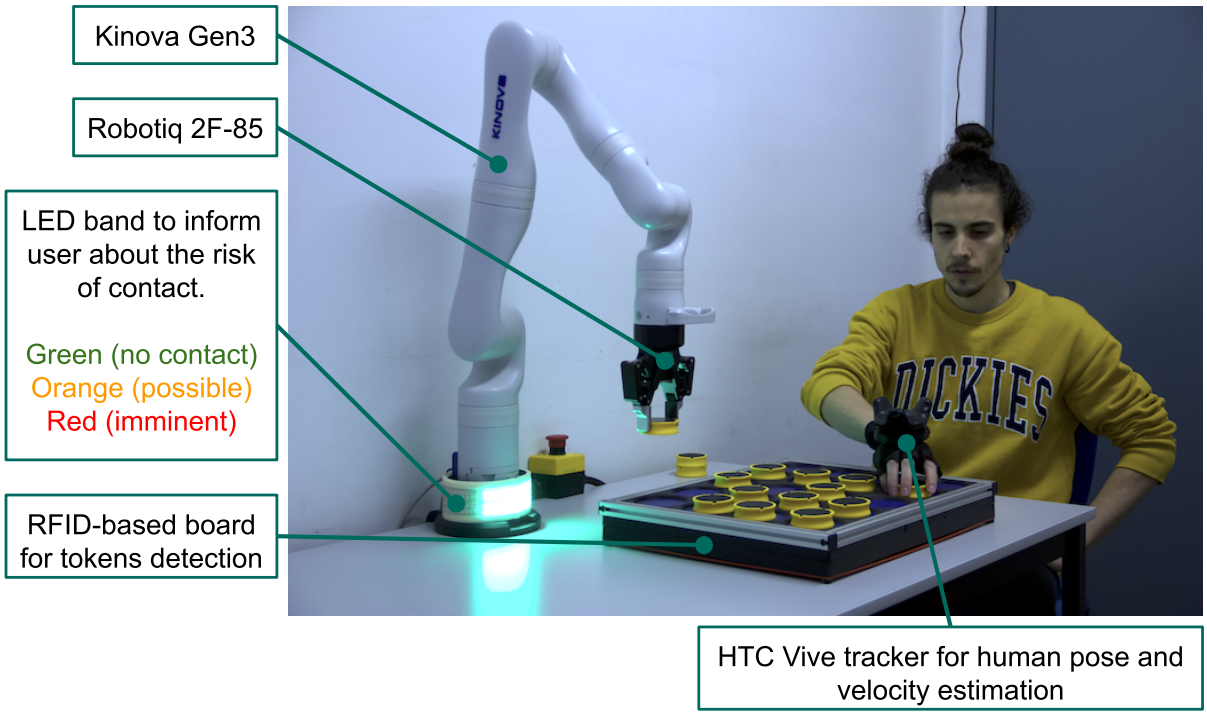


Figure 5.2: Setup of collaboratively filling a tray: example of an industrial kitting task used during the validation of this work.

of one of the experiments can be found in the additional material. This validation was meant to evaluate the ontology's capabilities to answer the set of competency questions proposed in Sec 5.3.1. Note that the design requirements of the ontology were those competency questions. Hence, answering them proves that the ontology was properly formalized, and that it meets the desired prerequisites. Specifically, the competency questions were contextualized by illustrating various situations extracted from the proposed collaborative scenario. For each situation, an OWL 2 DL knowledge base populated with the proper instances was used to answer the queries. Note that, as one can reason over OCRA using an inference engine (HermiT [Glimm et al., 2014]), this validates the consistency and coherence of the ontology.

To reliably compute the risk of human-robot collision, it was used the method proposed in Chapter 4, which computed the Time-To-Contact (TTC). Recall that TTC is the time that would take the robot's end effector and the human's hand to collide if they kept moving at the same relative velocity. Hence, the pose and velocity of the human was extracted from an HTC Vive tracker attached to the human's hand, and the measurements were taken at 100Hz. When the TTC was lower than a certain threshold, the robot stopped (high risk of collision). The medium degree of risk corresponded to when TTC was greater than the threshold and different to infinite. When TTC was infinite, meaning that there was no expected contact, the level of risk was low.

Instance name	**Ontological class**
Collaborative_workspace	Collaboration Place
Collaborative_workspace_role	Collaborative Place
CollaborativelyFillingATray	Event (later inferred as Collaboration)
FullTray	Goal
Human_operator	Physical Agent
Kinova_robot	Physical Agent
PickingAndPlacingTokensUntilFullTray	Plan
RFID_board	Designed Artifact
RFID_board_current_capacity	Available Capacity

Table 5.3: ABox overview to answer general competency questions about collaboration in Protégé. Note that the knowledge also comprised relations between the different instances, allowing to make inferences that were not originally asserted (e.g. that an event is indeed a collaboration).

Using the lights on the robot's base, the robot shared its interpretation of the collision's risk with the operator (Fig. 5.2). An RFID-based board was used for a fast and precise token-compartment detection. The experiment's software ran in a desktop PC with an Intel Core i7-7800X CPU (12x 3.50 GHz), a 32 GB DDR4 RAM, and an NVIDIA GeForce FTX 1080 Ti/PCLe/SSE2 GPU.

5.4.1 Filling a tray - Application ontology

In order to represent the knowledge of the proposed use case, some extra concepts were necessary. They were defined as either instances or specializations of DUL's classes. In the scenario of filling a tray, there were different objects: the robot, the human operator, the board (tray), the compartments, and the tokens. All of them were instances of different sub-classes of `PhysicalObject` in DUL: *'any Object (DUL) that has a proper space region'*. The robot and the human were defined as instances of `PhysicalAgent`, and the board, the compartments, and the tokens as instances of `DesignedArtifact`. For the board and the compartments, a new class was included: `AvailableCapacity`, defined as a `Quality` in DUL. This quality lets us capture the knowledge about whether a compartment or the tray is already filled or not.

5.4.2 Part 1 - Questions about collaboration

We deal here with the first five competency questions presented in Sec. 5.3.1. Once imported DUL and OCRA, all the entities that were involved in the collaboration are instantiated in the knowledge base (see Table 5.3). Note that the queries are presented in a description-logic-like syntax that is consistent to how the queries can be answered using a Protégé knowledge base.

Which and how many collaborations are running now? (Q1)

$$'is\ instance\ of'\ Collaboration$$

If the query holds, i.e. the knowledge base contains asserted or inferred instances of collaborative events, the answer contains all the possible values that make the query to be 'true'. In this case, there is one existent collaboration: 'collaboratively filling a tray'.

Which is the plan of a collaboration? (Q2)

$$'is\ plan\ executed\ in'\ value\ CollaborativelyFillingATray$$

The answer would contain all the plans that are executed in the collaborative event: 'picking and placing tokens until full tray'.

Which is the goal of a collaborative plan? (Q3)

$$'is\ component\ of'\ some\ (Plan\ and\ 'is\ plan\ executed\ in'\ value\ CollaborativelyFillingATray)$$

The answer would contain the goal to be achieved by executing the previously obtained collaborative plan: 'full tray'.

Are these agents collaborating? (Q4)

$$'is\ participant\ in'\ value\ CollaborativelyFillingATray\ and\ 'hasgoal'\ value\ FullTray\ and\ 'hasplan'\ value\ PickingAndPlacingTokensUntilFullTray$$

In this case, the answer contains that two different agents are collaborating: 'Kinova robot' and 'Human operator'.

Where is a collaboration happening? (Q5)

$$'is\ location\ of'\ value\ CollaborativelyFillingATray$$

The answer to this query says that the collaboration is happening at: 'Collaborative workspace'.

Instance name	Ontological class
Collaborative_workspace	Collaboration Place
Collaborative_workspace_role	Collaborative Place
CollaborativelyFillingATray	Event (later inferred as Collaboration)
CollaborativelyFillingATray_current_risk	Collaboration Risk
CollaborativelyFillingATray_directly_physical	Directly Physical Collaboration
CollaborativelyFillingATray_indirectly_physical	Indirectly Physical Collaboration
CollaborativelyFillingATray_non-physical	Non-physical Collaboration
FullTray	Goal
Human_operator	Physical Agent
Kinova_robot	Physical Agent
PickingAndPlacingTokensUntilFullTray	Plan
RFID_board	Designed Artifact
RFID_board_current_capacity	Available Capacity

Table 5.4: ABox overview to answer general competency questions about collaboration types and risks in Protégé. Note that the knowledge also comprised relations between the different instances, allowing to make inferences that were not originally asserted (e.g. that an event is indeed a collaboration).

5.4.3 Part 2 - Questions about collaboration types and risk

In this case, the Protégé knowledge base contained the same content as before plus some instances about the specific type of collaboration and the risk (see Table 5.4).

From the same collaboration event, different situations were extracted for each collaboration type and risk. Three different types of collaboration were considered corresponding to the ones defined in Sec. 5.3.4. Fig. 5.3 depicts a picture for each of the types. Regarding the collaboration risks, it was selected the risk of collision, which had three different levels: high, medium and low (see Fig. 5.4).

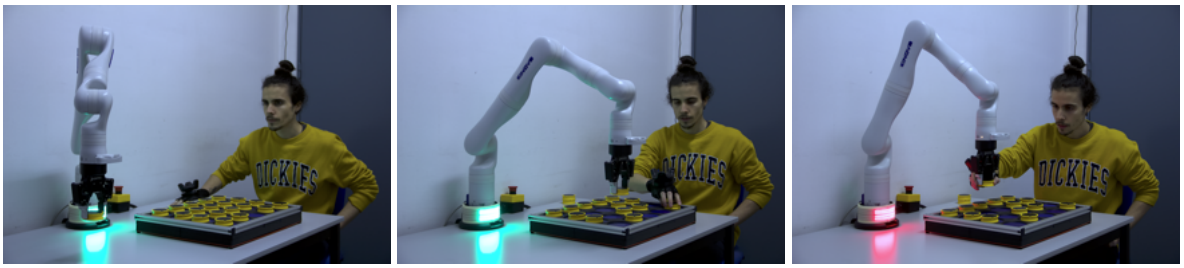
How is a collaboration classified? (Q6)

$$classifies\ value\ CollaborativelyFillingATray$$

The answer to this query says that the current type of collaboration is non-physical, but note that this value may change according to the cases depicted Fig. 5.3.

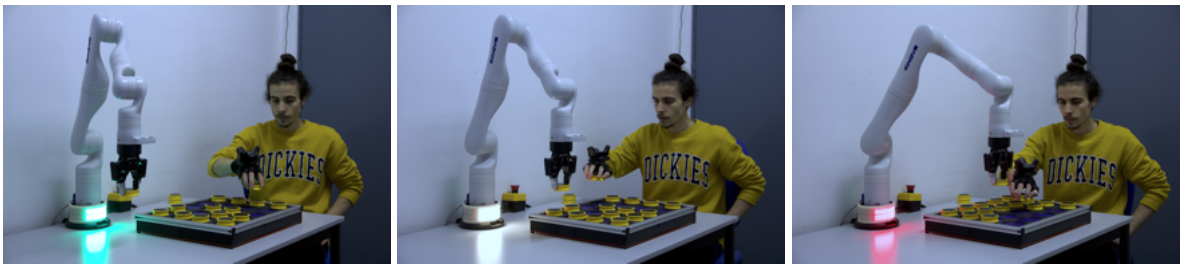
Which is the risk of a collaboration? (Q7)

$$'is\ instance\ of'\ CollaborationRisk\ and\ isQualityOf\ value\ CollaborativelyFillingATray\ and\ hasDataValue\ value\ 'HIGH\_RISK'$$



(a) (b) (c)

Figure 5.3: Collaboration types. (a) Non-physical collaboration: the robot selects the compartment to place a token and the human monitors how the robot does its part of the plan. (b) Indirectly physical collaboration: the human and the robot move to place a token in different compartments without exchanging forces. (c) Directly physical collaboration: the human moves the robot exchanging forces while the robot remains in admittance mode.



(a) (b) (c)

Figure 5.4: Collaboration risks. (a) Low risk - green light: there is not any potential detected collision. (b) Medium risk - orange light: the robot has detected a possible collision. (c) High risk - red light: the detected collision is imminent.

Note that the actual value of level of risk of a collaboration changes over time and is asserted as a data value. However, using the OWL 2 DL reasoners from Protégé one would always get the same answer, because they only work with classes and instances. In order to overcome this, and to avoid introducing other languages such as SPARQL [Pérez et al., 2006], SQWRL [O'Connor and Das, 2009], the data value of the current risk is restricted in the query. Hence, if the entity containing the current risk had the queried data value, it would be returned as a result. Otherwise, the result would be empty. In this case, the queried level is 'high' and there is a non-empty answer: 'Collaboratively filling a tray - current risk'.

5.4.4 Part 3 - Questions about adaptation

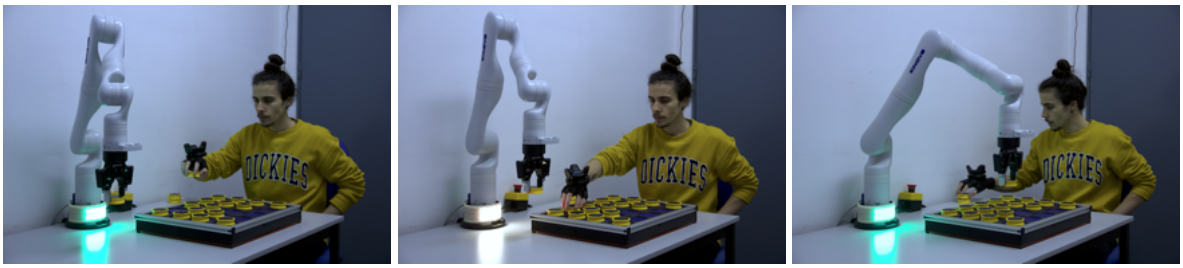
In this section, the remaining competency questions are answered. Once imported DUL and OCRA, all the entities that were involved in the adaptation event were instantiated in the

Instance name	**Ontological class**
Collaborative_workspace	Collaboration Place
CollaborativelyFillingATray	Collaboration
Full_compartment_adaptation	Event (inferred as Plan Adaptation)
Full_compartment_adaptation_final_plan	Plan
Full_compartment_adaptation_final_plan_execution	Event
Full_compartment_adaptation_initial_plan	Plan
Full_compartment_adaptation_initial_plan_execution	Event
FullTray	Goal
Human_operator	Physical Agent
Kinova_robot	Physical Agent
RFID_board	Designed Artifact
RFID_board_compartment_19	Designed Artifact
RFID_board_compartment_19_is_full	Situation
RFID_board_compartment_19_current_capacity	Available Capacity
RFID_board_current_capacity	Available Capacity

Table 5.5: ABox overview to answer competency questions about plan adaptation in Protégé. Note that the knowledge also comprised relations between the different instances, allowing to make inferences that were not originally asserted (e.g. that an event is indeed a plan adaptation).

knowledge base (see Table 5.5).

A new situation for the competency questions about adaptation is proposed. The robot modified its symbolic task plan and continued with the other free targets after the human filled one of the compartments the robot was meant to fill (see Fig. 5.5). Note that the robot's path planning was a simple point-to-point straight navigation.



(a) (b) (c)

Figure 5.5: Plan adaptation to unforeseen events. In the shown sequence, (a) the human and the robot move towards the same compartment; (b) then the human fills the compartment; (c) the robot adapts its plan and moves to another free compartment.

Which and how many plan adaptations are running now? (Q8)

$$'is\ instance\ of'\ PlanAdaptation$$

The answer contains all the possible values that make the query to be 'true'. In this case, there is one existent plan adaptation: 'full compartment adaptation'.

Which is/are the agent/s participating in the plan adaptation? (Q9)

$$'is\ participant\ in'\ value\ Full\_compartment\_adaptation$$

The answer says that there is one agent participating in the adaptation: 'Kinova robot'.

Why is an adaptation of an agent's plan happening? (Q10)

$$'is\ instance\ of'\ Situation\ and\ isPostconditionOf\ some$$
$$(Event\ and\ isPartof\ value\ Full\_compartment\_adaptation\ and\ executesPlan\ some$$
$$(Plan\ and\ isWorsePlanThan\ some\ Plan))$$

The answer states that there is a situation that triggered the adaptation: the compartment 19 was full. In this case, one could further ask for the details of the adaptation's cause, which are also represented using OCRA:

$$hasSetting\ value\ RFID\_board\_compartment\_19\_is\_full\ and\ hasQuality\ some$$
$$(AvailableCapacity\ and\ hasDataValue\ value\ 0)$$

The query holds since the current situation is setting of compartment nineteen, whose available capacity is zero, thus the obtained answer is: 'RFID board compartment 19'. This indicates that the compartment is full, the reason why the robot adapts its plan.

Which is the plan before and after the adaptation? (Q11)

There are two queries to do in this case, one per each of the plans.

$$isPlanExecutedIn\ some\ (Event\ and\ isPartOf\ value\ Full\_compartment\_adaptation)$$
$$and\ isWorsePlanThan\ some\ Plan$$

This would return the initial plan (filling compartment 19).

$$isPlanExecutedIn\ some\ (Event\ and\ isPartOf\ value\ Full\_compartment\_adaptation)$$
$$and\ isBetterPlanThan\ some\ Plan$$

The answer to this query would return the final plan (filling compartment 4).

Which is the goal of the agent involved in the adaptation that is also the goal to be achieved by both the old and the new plan? (Q12)

$$isComponentOf\ value\ Full\_compartment\_adaptation\_initial\_plan\ and\ isComponentOf\ value\ Full\_compartment\_adaptation\_final\_plan\ and\ isGoalOf\ some\ (Agent\ and\ isParticipantIn\ value\ Full\_compartment\_adaptation)$$

The answer contains that the goal is: 'Full tray'.

5.5 Validation II - Limit cases evaluation

This validation aims to study the robustness of the proposed ontological model, analyzing OCRA's performance in several limit cases of the formalization. Particularly, a set of examples of `Collaboration` and `Plan Adaptation` is proposed that contain incongruent or incomplete axioms (see Tables 5.6 and 5.7). We explore how the formal definitions in FOL and OWL 2 DL behave in these cases, observing whether OCRA is able to exclude or not the incorrect instances. The results show that the formal definitions within OCRA indeed exclude them in most of the cases. This proves the strength of our formal model in situations where it might be unclear whether an event is or not a `Collaboration` or a `Plan Adaptation`.

Case description	**Classification**	**FOL**	**OWL 2 DL**
An agent (robot) during the execution of plan (o) and due to a situation (s), realizes that there is a plan (n) that has the same goal and is better than the initial plan (o). However, the agent continues executing (f) the old plan (o).	Not a plan adaptation since the agent still executes the initial plan.	executesPlan(f,n) $\wedge$ $\neg$executesPlan(f,o) are violated, thus the case is excluded by the definition in FOL.	executesPlan(f,n) $\wedge$ $\neg$executesPlan(f,o) are violated, thus the case is excluded by the definition in OWL 2 DL.
An agent (robot) during the execution of a plan (o) decided to change and execute another plan (n) that has the same goal. Nevertheless, the new plan (n) is not better than the original plan (o) due to the actual situation (s) realized after the execution of an initial part (i) of the original plan (o).	Not a plan adaptation since no situation makes the new plan a better one.	betterPlan(s,n,o) is violated, thus the case is excluded by the definition in FOL.	isBetterPlanThan(n,o) is violated, thus the case is excluded by the definition in OWL 2 DL.

Table 5.6: Ontology robustness evaluation of the formalization of Plan Adaptation.

Case description	Classification	FOL	OWL 2 DL
A human (h) and a robot share the plan and the goal during the plan's execution (e) but the human performs no activity.	Not a collaboration since only one of the agents (the robot) is active.	dul.hasParticipant(e,h) is violated, thus the case is excluded by the definition in FOL.	dul.hasParticipant(e,h) is violated, thus the case is excluded by the definition in OWL 2 DL.
A human and a robot share the plan (p) and the goal during an event (e) in which they both perform activities but without executing the shared plan.	Not a collaboration since the event does not execute the shared plan.	executesPlan(e,p) is violated, thus the case is excluded by the definition in FOL.	executesPlan(e,p) is violated, thus the case is excluded by the definition in OWL 2 DL.
A human and a robot have the same plan (p) and goal (g) during the time that both execute the plan, but the plan's goal is different from the shared goal.	Not a collaboration since the agents execute a plan to achieve a goal that is not shared.	dul.hasComponent(p,g) is violated, thus the case is excluded by the definition in FOL.	dul.hasComponent(p,g) is violated, thus the case is excluded by the definition in OWL 2 DL.
A human and a robot share the plan during the time that its execution lasts (t). They also share the goal (g) but not during the whole execution, because the robot (r) changes its goal at some point.	Not a collaboration since the human and the robot do not share the goal during the whole execution of their plan.	hasGoal(r,g,t) is violated, thus the case is excluded by the definition in FOL.	hasGoal(r,g) holds some time, thus the case is not excluded by the definition in OWL 2 DL. It might be solved by 'reification', introducing several relations $hasGoal\_tp(r, g)$, one for each instant (tp) in which the relation holds. This solution is activity-dependent so we do not present it in the general definition.

Table 5.7: Ontology robustness evaluation of the formalization of Collaboration.

5.6 Discussion

This chapter proposed OCRA, an Ontology for Collaborative Robotics and Adaptation. In harmony with the findings of the initial chapters, it has been built around two main concepts: collaboration, and plan adaptation. The proposed definitions included in the ontological model are consistent with the state of the art, and they were formalized in FOL, to take advantage of its expressiveness, and in OWL 2 DL for practical computational purposes. The ontological theory was qualitatively validated in a realistic case study in which a human and a robot shared

the execution of a task. First, the capability of the ontology to answer a set of competency questions in a contextualized scenario was assessed. Second, it was discussed how the formalization would work in some limit cases in which we purposely defined wrong instances of `Collaboration` and `Plan Adaptation`. Using OCRA, robots can formally represent, reason about, and recognize adaptive and collaborative events in unstructured collaborative robotic scenarios.

This work is a step forward to more reliable collaborative robots, and also to enhance the interoperability and reusability of the terminology in this domain. It remains open though how this work can contribute to foster the development of ontology-based explainable robots. The introduction in Chapter 1 already discussed that explainable agency requires functional abilities such as: reporting the actions robots executed (e.g. collaboration with humans), explaining how actual events diverged from what was planned and how robots adapted to it (i.e. plan adaptation), and explain decisions made during plan generation (comparing alternatives). The ontological model proposed in this chapter lets robots represent knowledge related to the first two functional abilities. However, further research needs to be conducted on how robots can manipulate such knowledge to construct explanations of collaborative and adaptive experiences (see Chapter 6). Furthermore, a more comprehensive model would be needed to generate ontology-based explanations about decisions made during plan generation and comparison, which is discussed in Chapter 7.

chapter six

Robots narrating collaborative and adaptive experiences

> *..the universe is made of stories, not of atoms..*
>
> **— Muriel Rukeyser**
> (The speed of darkness)

This thesis aims to explore the use of ontologies as an integrative framework for explainable robotics, advocating for the storage of ontology-based robot episodic memories for latter retrieval and explanation construction. Chapter 1 discussed that explainable robots would require functional abilities such as: producing reports of their executed actions (e.g. collaborative experiences), and explaining how they adapted to unexpected changes (e.g. adaptive events). In this regard, the validation with users from Chapter 3 disclosed interesting insights about the potential benefits of explainable robots in collaborative and adaptive scenarios. Using a simple LED armband already seemed to help users to understand why the robot was not adapting accordingly to the expected collaboration, which boosted mutual understanding. We thought that more comprehensive robot knowledge about collaborative and adaptive events would lead to greater advancements. Hence, a novel ontological model for collaborative robotics and adaptation was introduced in Chapter 5, allowing robots to represent knowledge about their collaborative and adaptive experiences. Expanding upon those prior contributions, the current chapter deals with investigating how robots might leverage such knowledge to construct and communicate explanations of what robots know

about those experiences. For that, it is proposed a sound methodology that integrates three main elements: first, the ontology for collaborative robotics and adaptation presented in Chapter 5, which models the domain knowledge; second, an episodic memory for time-indexed knowledge storage and retrieval; third, a novel algorithm to extract the relevant knowledge and generate textual explanatory narratives. The algorithm produces three different types of outputs, varying the specificity, for diverse uses and preferences. A pilot study was conducted to assess the usefulness of the narratives, yielding promising results in fostering a shared understanding between humans and robots. Finally, the chapter discusses how the methodology can be generalized to other ontologies and experiences. Note that since the approach is time-sensitive (the knowledge is episodic), it can be used to narrate details of short and also long-term robot past experiences. This chapter marks the foundational stone for further advancements in ontology-based explainable robotics within this thesis.

6.1 Motive

The development of applications where humans and robots collaborate triggers the appearance of several issues such as those related to trustworthiness between the collaborative agents. For proper cooperation, mutual understanding of the ongoing events and communication between teammates become essential [Yuan et al., 2022]. In this regard, narratives seem to help with understanding agents' actions [Carr, 2008]. Hence, collaborative robots could narrate what they know of their experiences, i.e., collaborations and plan adaptations, to be more understandable. Those robot narratives may boost explainable agency (i.e., explaining the reasoning of goal-driven agents and robots), which has recently gained significant momentum [Anjomshoae et al., 2019, Chakraborti et al., 2020]. Robotic tasks may involve several events and a lot of contextual knowledge. Hence, time-indexed narratives of events (i.e., narrating events when they occur) make more sense in robotics than in other artificial intelligence tasks (e.g. classification), where single post hoc and time-independent narratives or explanations might suffice.

Langley et al. [Langley et al., 2017], discussed the need for three elements of explainable agency: a representation of the domain knowledge, an episodic memory to store the knowledge, and the ability to access and retrieve that knowledge to generate explanations. Episodic memory is the collection of past personal experiences that occurred at particular times and places [Tulving, 1972]. Beetz et al. [Beetz et al., 2018], presented the second generation of KnowRob, a knowledge-based framework for robotics, which includes formal domain ontologies, and narrative-enabled episodic memory (NEEM) storage and retrieval. NEEMs may be useful for generating human understandable explanatory narratives, but this is still

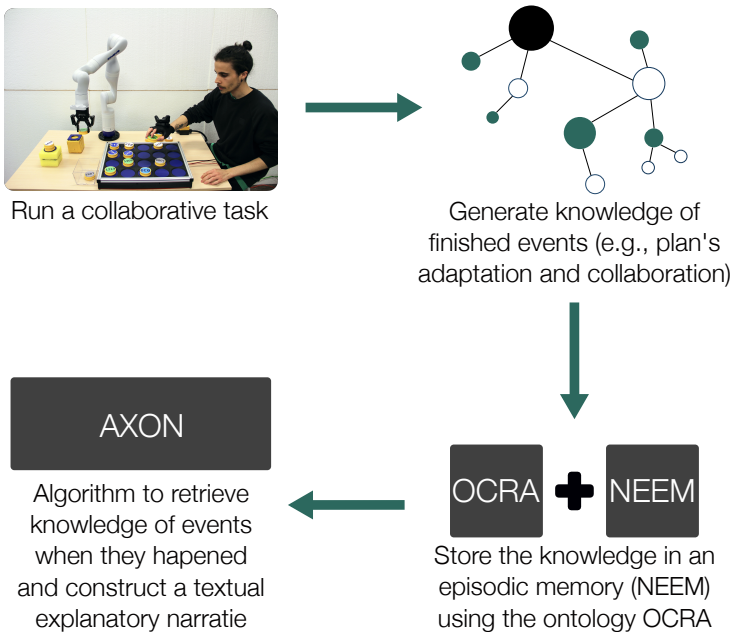


Figure 6.1: Overview of the methodology for the generation of explanatory ontology-based narratives for collaborative robotics and adaptation (XONCRA).

unexplored, especially in the collaborative robotics and adaptation domain. Hence, this chapter proposes a methodology (see Fig. 6.1) for the generation of eXplanatory Ontology-based Narratives for Collaborative Robotics and Adaptation (XONCRA). The proposed methodology and its evaluation are designed to address the following research questions:

- **RQ1 -** How can robots construct the narrative of their collaborative and adaptive events (experiences)?
- **RQ2 -** How does the narratives' specificity affect the users perceived usefulness of the received information?

6.2 Related work

We found great inspiration in the narrative and storytelling literature. Labov et al. [Labov and Waletzky, 1997], defined a narrative *'as a way of recounting past events, in which the order of narrative clauses matches the order of events as they occurred'*. Carr [Carr, 2008], stated that to provide explanatory information, a narrative should contextualize the agent's experiences in time. Both works emphasized the importance of the time when the events occurred, reinforcing

the need for episodic memory. Narratives have already been an applied for robot task plans' verbalization [Rosenthal et al., 2016, Flores et al., 2018, Canal et al., 2022].

A sound approach to represent domain knowledge is to use representation formalisms such as ontologies. The 1872–2015 IEEE Standard Ontologies for Robotics and Automation [Schlenoff et al., 2012] and the 1872.2-2021 IEEE Standard for Autonomous Robotics Ontology [Gonçalves et al., 2021b] were developed to become references for knowledge representation in the domain. Indeed, the use of ontologies has spread to several robotic sub-domains such as service and assistive robotics [Olszewska et al., 2017, Fiorini et al., 2017, Olivares-Alarcos et al., 2019a]. Some examples are manufacturing and collaborative robotics [Stenmark and Malec, 2015, Balakirsky, 2015, Chen et al., 2021, Borgo et al., 2019b, Sampath Kumar et al., 2019, Umbrico et al., 2020, Olivares-Alarcos et al., 2022], robot co-design [Ramos et al., 2018b, Ramos et al., 2018a], and service and general purpose robots [Beetz et al., 2018, Bruno et al., 2019a, Beßler et al., 2020b]. All these works are steps towards the harmonization and formalization of the knowledge in the robotics domain. Hence, they have the potential to play a major role in the explainable agency.

The notion of episodic memory was first introduced in a classical work by Tulving as the collection of past personal experiences that occurred at particular times and places [Tulving, 1972]. Its essence lies in the conjunction of three concepts: self, autonoetic awareness, and subjectively sensed time [Tulving, 2002]. Beetz et al. [Beetz et al., 2018], introduced a knowledge-based framework for robots that includes an episodic memory, the narrative-enabled episodic memory (NEEM). It consists of the NEEM experience (low-level time-indexed information) and the NEEM narrative (symbolic descriptions, e.g., goals, states, etc.). NEEMs have already been used in human-robot interaction [Bartels et al., 2019], and robot learning [Bozcuoğlu et al., 2019]. Nevertheless, their role in the generation of robot textual narratives still remains unexplored.

In the literature, several authors worked on automatic text generation using knowledge modeled in OWL (Web Ontology Language) or RDF (Resource Description Framework) [Androutsopoulos et al., 2013, Nguyen et al., 2019, Ngonga Ngomo et al., 2019, Dalianis and Hovy, 1996]. Although inspiring, none of those works discussed the generation of different types of texts based on the preferred specificity. Furthermore, in ours, the target knowledge to be included in the textual narratives is automatically retrieved, while the others just assumed that the knowledge atoms or tuples were given.

6.3 Explanatory ontology-based narratives for collaborative robotics and adaptation

6.3.1 Preliminary notation

Let's assume countable pairwise disjoint sets N_C, N_P, and N_I of class names, property names, and individuals, respectively. The standard relation `rdf:type`, which relates an individual with its class, is abbreviated as `type` and included in N_P. A knowledge graph $\mathcal{G}$ is a finite set of triples of the form $\langle s, p, o\rangle$ (subject, property, object), where $s \in N_I$, $p \in N_P$, $o \in N_I$ if $p \neq type$, and $o \in N_C$ otherwise. The semantic knowledge of an episodic memory can be seen as a time-indexed knowledge graph $\mathcal{G}_\mathcal{T}$, which is a finite set of tuples of the form $\langle s, p, o, t_i, t_f\rangle$, where $t_i, t_f \in \mathbb{R} > 0$, and denote the time interval (initial and final time) in which the triple $\langle s, p, o\rangle$ holds. Knowledge graphs commonly comply with the open-world assumption, thus, non-asserted triples are unknown instead of false. For this reason, the second version of the Web Ontology Language (OWL 2) allows to make explicit negative properties assertions: $\langle s, p, o\rangle$ is $false$.[1] Hence, in $\mathcal{G}_\mathcal{T}$ one may store, for instance, that during an interval of time, $\langle t_i, t_f\rangle$, an event e is not an instance of the class `Collaboration`: $\langle e, type, Collaboration, t_i, t_f\rangle$ is $false$. In this work, querying the $\mathcal{G}_\mathcal{T}$, we build what we called 'narrative tuples' of an instance event, $\mathcal{T}_e$: $\langle s, p, o, t_i, t_f, sign\rangle$, where $sign$ indicates whether the time-indexed triple comes from a positive or negative assertion.

6.3.2 NEEMs for collaborative robotics and adaptation

The proposed methodology incorporates a knowledge-based episodic memory for collaborative robots that adapt to unstructured scenarios. It consists of the integration of an ontology for collaborative robotics and adaptation (OCRA) [Olivares-Alarcos et al., 2022], into the NEEMs ecosystem of Knowrob [Beetz et al., 2018]. It allows robots to represent time-indexed knowledge of their collaborations and adaptations, store it and retrieve it for a later generation of textual explanatory narratives.

Background on OCRA

The ontology, introduced in Chapter 5), was developed to enhance the reusability of the domain's terminology, and to allow robots to formalize and reason about two main concepts: collaboration and plan adaptation. `Collaboration` is defined as *'an event in which two or*

[1] www.w3.org/2007/OWL/wiki/FullSemanticsNegativePropertyAssertions

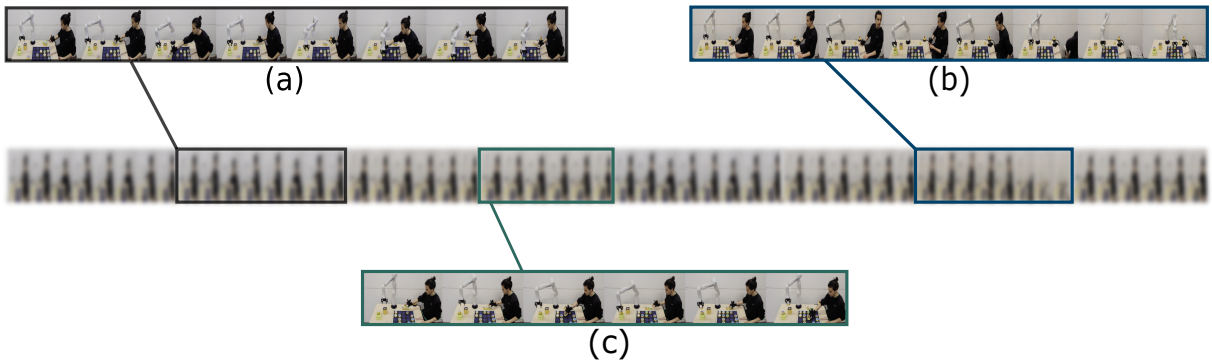


Figure 6.2: Visualization of a recorded NEEM of a prototypical collaborative kitting task (filling a tray) with different episodes within it (a, b, and c).

more agents share a goal and a plan to achieve the goal, and execute the plan while interacting'. `Plan Adaptation` is *'an event in which one (or more) agent, due to its evaluation of the current or expected future state, changes its current plan while executing it, into a new plan, in order to continuously pursue the achievement of the plan's goal'*. Considering these definitions, narratives of `Collaborations` shall include knowledge about the shared plan and goal, and the agents executing the plan. Meanwhile, narratives of `Plan Adaptations` must contain details regarding the initial and new plans, the situation triggering the adaptation, and the involved agent. In this chapter, the OCRA's formalization in OWL 2 DL, a description logic version of OWL 2, is used.

Background on NEEMs

For every activity the robot (agent) performs, observes, or prospects, it can create an episode and store it in its memory. An episode is best understood as a video recording that the robot makes of the ongoing activity (see Fig. 6.2). In addition, those videos are enriched with a very detailed log of the actions, motions, their purposes, effects, and the agent's sensor information during the activity. The episodic memories created by Knowrob are named narrative-enabled episodic memories (NEEMs). A NEEM consists of the NEEM experience and the NEEM narrative. The NEEM experience captures low-level data such as the agent's sensor information, e.g. images and forces, and records of poses of the agent and its detected objects. NEEM experiences are linked to NEEM narratives, which are logs of the episode described symbolically. These narratives contain information regarding the tasks, the context, intended goals, observed effects, etc. In this chapter, the focus is on the NEEM-narrative, since the aim is to explain the symbolic understanding that the robot has of its experiences. A detailed overview of NEEMs can be found in the NEEM Handbook [Beetz et al., 2020].

Integration

NEEMs are modeled using OWL 2 DL ontologies built upon the DOLCE+DnS Ultralite (DUL) foundational ontology [Borgo et al., 2021], the same upper-level ontology that OCRA relies on. OCRA was integrated into Knowrob's and NEEMs' ecosystem without causing any ontological inconsistency. The knowledge base is accessible to the robot through a prolog-based service implemented as a ROS (Robot Operating System) package: rosprolog.[2] Here it was implemented a novel ROS package (know-cra) in which OCRA is integrated into Knowrob's framework. This implementation is publicly available on a GitHub repository,[3] and illustrates how to load and use OCRA, and some instantiated use cases, with Knowrob. Furthermore, the shared code also includes examples of manipulating recorded NEEMs.

6.3.3 AXON - An algorithm for explanatory ontology-based narratives

AXON is the major theoretical contribution of this chapter, a novel algorithm that extracts knowledge about target experiences or events from episodic memories, and uses it to construct textual explanatory narratives. The algorithm leverages the structure of ontological episodic knowledge, assuming that knowledge connected to a robot experience is semantically relevant to narrate it. The time-indexed knowledge graph $\mathcal{G}_{\mathcal{T}}$ stored in the episodic memory is the first algorithm's input. Furthermore, AXON takes three more inputs: the ontological class (or classes) of the events to narrate, the temporal locality (time interval of the events of interest), and the level of specificity. Although our focus is on narratives about `Collaborations` and `Plan adaptations`, AXON is general enough to work with other OWL 2 DL ontologies and classes, as it is discussed in Sec. 6.5. There are three different narrative types, depending on the selected specificity. In this work, specificity refers to the amount of detail used to construct the textual narrative, more precisely, the number of knowledge tuples. This section first introduces the main algorithm (see Alg. 3), and then explains its three major routines: Retrieve Instances With Time Interval, Retrieve Narrative Tuples, and Construct Narrative. An implementation of the algorithm and an example of use can be found in an online repository.[4]

AXON first retrieves a set $\mathcal{I}_{\mathcal{T}}$ of tuples $\langle e, t_i, t_f \rangle$, containing the event instances e of the provided classes $\mathcal{C}$ whose time interval (t_i, t_f) exists, at least partially, within the temporal locality (L_i, L_f) (line 2). Second, based on the specificity S, the algorithm retrieves a set of knowledge tuples $\mathcal{T}_e$ related to each instance (line 4). Third, an explanation $\mathcal{E}_e$ for every instance is constructed using their respective tuples (line 5). Finally, the algorithm

[2] www.github.com/knowrob/rosprolog
[3] https://github.com/albertoOA/know_cra
[4] https://github.com/albertoOA/explanatory_narratives_cra

Algorithm 3: AXON

Input: Episodic memory ($\mathcal{G}_\mathcal{T}$), events to narrate ($\mathcal{C}$), temporal locality (L_i, L_f), specificity (S)
Output: Narrative ($\mathcal{E}$)

1 $\mathcal{E} \leftarrow \emptyset$
2 $\mathcal{I}_\mathcal{T} \leftarrow$ RetrieveInstancesWithTimeInterval($\mathcal{G}_\mathcal{T}$, $\mathcal{C}, L_i, L_f$)
3 **foreach** $\langle e, t_i, t_f \rangle \in \mathcal{I}_\mathcal{T}$ **do**
4 $\quad \mathcal{T}_e \leftarrow$ RetrieveNarrativeTuples($\mathcal{G}_\mathcal{T}, \langle e, t_i, t_f \rangle, S$)
5 $\quad \mathcal{E}_e \leftarrow$ ConstructNarrative($\mathcal{T}_e$)
6 $\quad \mathcal{E} \leftarrow \mathcal{E} \cup \mathcal{E}_e$
7 **end**

concatenates the new explanation to the set of explanations $\mathcal{E}$ (line 6).

Retrieve instances with time interval routine

Given a time-indexed knowledge graph $\mathcal{G}_\mathcal{T}$, an ontological existing class or a set of them, $\mathcal{C} \subset N_C$, and a time interval $\langle L_i, L_f \rangle$, this routine retrieves a set $\mathcal{I}_\mathcal{T}$ containing all the time-indexed instances $\langle e, t_i, t_f \rangle$ of the given classes such that $\forall \langle e, t_i, t_f \rangle \in \mathcal{I}_\mathcal{T} \rightarrow \exists c \in \mathcal{C} \wedge \langle e, type, c, t_i, t_f, sign \rangle \wedge \langle t_i, t_f \rangle \cap \langle L_i, L_f \rangle$. Some examples of instances of events to narrate with their time interval may be the following:

$\langle Event\_15, 100.0, 142.0 \rangle$,
$\langle Event\_27, 200.0, 240.0 \rangle$.

Retrieve narrative tuples routine

Given $\mathcal{G}_\mathcal{T}$, an instance event e to narrate with the time interval in which it exists $\langle t_i, t_f \rangle$, and the specificity level S, this routine retrieves all the relevant tuples, $\langle s, p, o, t_i, t_f, sign \rangle$, to construct the narrative. The first level of specificity can be considered as a baseline and only returns tuples containing the class c of each instance: $\langle e, p, c, t_i, t_f, sign \rangle \in \mathcal{G}_\mathcal{T} \wedge p = type$. In the second level, the algorithm adds all the tuples in which the instance e is related to an object o through any property different to `type`: $\langle e, p, o, t_i, t_f, sign \rangle \in \mathcal{G}_\mathcal{T} \wedge p \neq type$. Finally, the third level adds all the tuples in which the objects o from the second level are related to other objects o_x: $\langle o, p_x, o_x, t_{ix}, t_{fx}, sign_x \rangle \in \mathcal{G}_\mathcal{T} \wedge \langle t_i, t_f \rangle \cap \langle t_{ix}, t_{fx} \rangle$. As robots' experiences are tied to a time frame, the search was restricted to tuples whose time interval $\langle t_{ix}, t_{fx} \rangle$ intersected the time interval of the instance $\langle t_i, t_f \rangle$. This aimed to avoid retrieving tuples that were irrelevant to the narrative of the instance e. Furthermore, if a tuple or its inverse already exists in the retrieved set $\mathcal{T}_e$, it is not added. Note that the retrieved tuples for each level are also included in upper levels (e.g.,

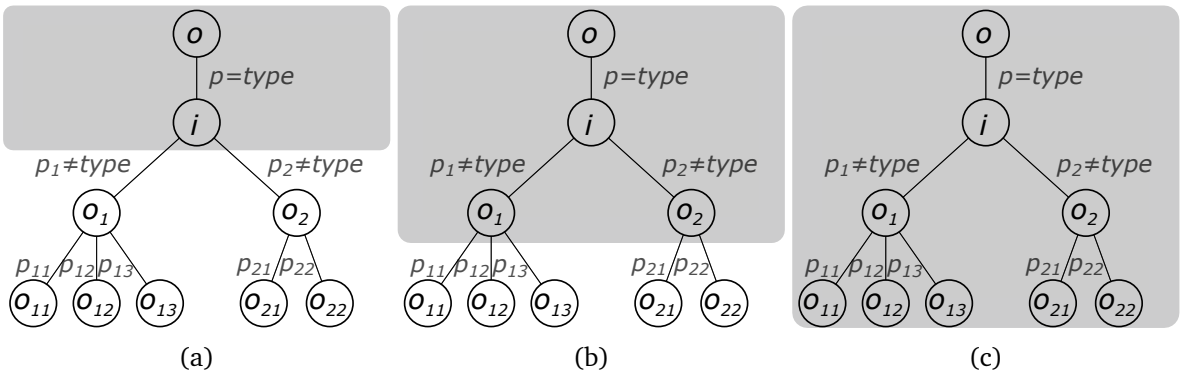


Figure 6.3: Graphical representation of the different levels of specificity (S) and their respective depth in the knowledge graph. (a) $S = 1$, (b) $S = 2$, and (c) $S = 3$.

the tuples from the first level are also returned in the second and the third). In an intuitive way, this would be equivalent to going deeper into the knowledge graph (see Fig. 6.3). Using as an example the task of filling a tray (from Chapter 5), some instances of the retrieved narrative tuples $\mathcal{T}_e$ of an event are:

$\mathcal{T}_{e_1}$. ⟨Robot, hasPlan, Place Tokens By Color, 200.0, 240.0, positive⟩,
$\mathcal{T}_{e_2}$. ⟨Event_27, hasParticipant, Robot, 200.0, 240.0, positive⟩,
$\mathcal{T}_{e_3}$. ⟨Place Tokens By Color, isPlanOf, Human, 200.0, 240.0, negative⟩,
$\mathcal{T}_{e_4}$. ⟨Human, isParticipantIn, Event_27, 200.0, 240.0, positive⟩,
$\mathcal{T}_{e_5}$. ⟨Human, type, Physical Agent, 1.0, 1000.0, positive⟩.

Construct narrative routine

Given the narrative tuples $\mathcal{T}_e$ of an instance event to narrate e, this routine constructs the final explanatory narrative following a set of rules: **casting, clustering, ordering, and grouping**. These rules, proposed by Dalianis et al. [Dalianis and Hovy, 1996], define the aggregations that humans usually do in natural language.

Casting consists of homogenizing all the properties used in the tuples. First, making sure that in all the tuples $\mathcal{T}_e$ concerning the target instance e ($\mathcal{T}_{e_2}$ and $\mathcal{T}_{e_4}$ in the previous example), e acts as the subject of the tuple. Hence, when $\mathcal{T}_e$ contains a tuple in which e acts as the object, $\langle s, p, e, t_i, t_f, sign\rangle \in \mathcal{T}_e$, the algorithm inverts the tuple to: $\langle e, p^{-1}, s, t_i, t_f, sign\rangle$, where p^{-1} is the inverse property of p. In the tuples shown before, the tuple $\mathcal{T}_{e_4}$ containing the property `isParticipantIn` would be changed using its inverse `hasParticipant`. Once this is done, all the tuples regarding e are added to the set of cast tuples $\mathcal{T}_{e_{Cast}}$. The second step in casting involves the tuples not concerning e ($\mathcal{T}_{e_1}$, $\mathcal{T}_{e_3}$ and $\mathcal{T}_{e_5}$), ensuring that each tuple's property is consistent with the properties already existent in the cast tuples. Otherwise, the tuple is inverted

before adding it to $\mathcal{T}_{e_{Cast}}$. In the example, $\mathcal{T}_{e_1}$ is added to $\mathcal{T}_{e_{Cast}}$ (following the order), thus, $\mathcal{T}_{e_3}$ needs to be inverted before added.

Then the routine **clusters** all the tuples $\langle s, p, o, t_i, t_f, sign\rangle$ that share the subject s. Therefore, when generating the narrative, all the information about a specific subject will appear together. In the example, $\mathcal{T}_{e_2}$ and the inverted $\mathcal{T}_{e_4}$, and the inverted $\mathcal{T}_{e_3}$ and $\mathcal{T}_{e_5}$ would be clustered.

Next, the tuples are **ordered**: externally and internally. The external ordering consists in ordering the subjects from more information (more tuples) to less. This rule has one exception, the information about the target instance is always at the top front of the list. The internal ordering ensures that the tuples with the property $p = type$ are at the front of the list for each subject. In the example, after applying all these rules the set of tuples would change to:

$\mathcal{T}'_{e_1}$. ⟨Event_27, hasParticipant, Robot, 200.0, 240.0, positive⟩,
$\mathcal{T}'_{e_2}$. ⟨Event_27, hasParticipant, Human, 200.0, 240.0, positive⟩,
$\mathcal{T}'_{e_3}$. ⟨Human, type, Physical Agent, 1.0, 1000.0, positive⟩,
$\mathcal{T}'_{e_4}$. ⟨Human, hasPlan, Place Tokens By Color, 200.0, 240.0, negative⟩,
$\mathcal{T}'_{e_5}$. ⟨Robot, hasPlan, Place Tokens By Color, 200.0, 240.0, positive⟩.

Finally, the tuples are **grouped** into a sentence, constructing the final textual narrative $\mathcal{E}_e$. First, the tuples with the same subject, property, interval, and sign are joined (object grouping). Hence, if there are two tuples: $\langle s, p, o_a, t_i, t_f, sign\rangle$ and $\langle s, p, o_b, t_i, t_f, sign\rangle$, the algorithm joins them to: $\langle s, p, o_a\ and\ o_b, t_i, t_f, sign\rangle$. In the example tuples, $\mathcal{T}'_{e_1}$ and $\mathcal{T}'_{e_2}$ would be joined into: ⟨Event_27, hasParticipant, Robot and Human, 200.0, 240.0, positive⟩. Second, the tuples for each subject are joined into separated sentences (subject grouping) considering their sign and using the conjunction 'and' and the propositions 'from' and 'to'. The adverb 'not' is included before the property when generating the text of a negative assertion. Furthermore, it is excluded the time interval of a tuple if it was equal to the time interval in which the instance exists. The names of properties, classes, and instances are kept, only the property 'type' is changed to 'is a type of'. The final narrative for the ongoing example would be:

'Event_27' has participant 'Robot and Human' from 200.0 to 240.0. 'Human' is a type of 'Agent' from 1.0 to 1000.0 and (not) has plan 'Place Tokens By Color'. 'Robot' has plan 'Place Tokens By Color'.

6.4 Validation: Setting the methodology to work

6.4.1 Collaborative task: filling a tray with tokens

The validation of XONCRA was contextualized in a lab mock-up of a real task, where a robot and a human shared the task of filling the compartments of a tray/board (see Fig. 6.4). The task's objective was to obtain a tray full of tokens. The specific order changes to create different

tasks (e.g., tokens are sorted by color, in ascending numerical order, etc). When a token was not useful to accomplish the task's goal (e.g., compartments for that color are already filled), it was discarded. The risk of human-robot collision was computed using the pose and velocity extracted from an HTC Vive tracker attached to the human's hand using the Time-To-Contact (TTC) method presented in Chapter 4.

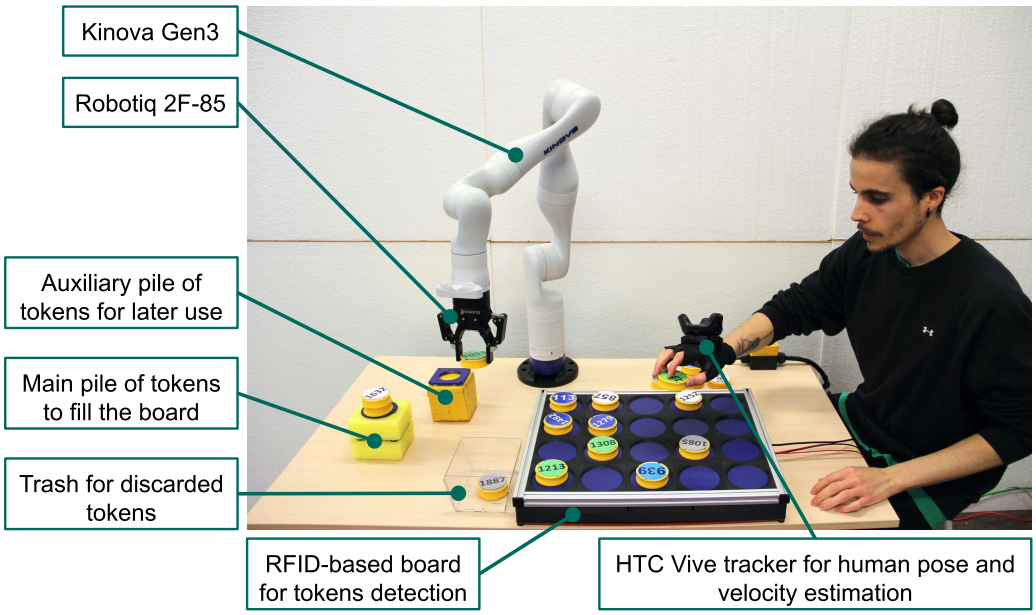


Figure 6.4: Setup of filling a tray, the validation task.

6.4.2 Robot experiences about collaboration and adaptation

Following the schema described in the proposed methodology XONCRA (see Fig. 6.1), RQ1 is addressed. The first step is to run executions, twelve in this case, of the validation task. Those executions were designed to showcase diverse situations of collaborations and adaptations according to how they are defined in OCRA. Hence, varying their main elements: the goal, the plan, and the workload distribution between the human and the robot. In order to ensure a curated knowledge base, the knowledge tuples involved in those executions were manually stored into a single NEEM after recording videos of the executions. From now on, we will refer to that NEEM as *validation NEEM*.

The twelve events included three cases of collaboration, six robot plan adaptations, and three other situations with non-collaboration. According to OCRA's definitions, in the collaborations, the human and the robot shared the goal (e.g. full board with tokens in columns ordered by color) and the plan, and both of them participated in accomplishing the goal. In the adaptations, the robot stopped executing a plan due to an unexpected situation and started executing a

Name in the NEEM	Case description
Event 28	An example of collaboration, the shared goal was to have a full board with tokens with odd numbers.
Event 30	An example of collaboration, the shared goal was to have a full board with tokens in ascending order.
Event 33	An example of collaboration, the shared goal was to have a full board with tokens by color in columns.
Event 9	An example of non-collaboration, the human stopped participating in the event to start taking notes.
Event 15	An example of non-collaboration, the human stopped participating in the event to leave the workspace.
Event 27	An example of non-collaboration, the human stopped executing the shared plan (filling in ascending order) to start executing a different plan (filling by colors in columns).
Event 39	An example of plan adaptation, the robot issued a safety stop due to a high risk of collision.
Event 43	An example of plan adaptation, the robot discarded the token to the trash because the number on the token was too small according to the current tokens on the board. The human has placed a token on the board that triggered the adaptation.
Event 49	An example of plan adaptation, the robot changed its target compartment because the human filled it.
Event 51	An example of plan adaptation, the robot placed the token on the auxiliary pile for later use because the target compartment is busy with an incorrect token. Note that the human is expected to pick and place the incorrectly placed token (freeing the compartment) because the robot cannot reach the pose where it should go.
Event 59	An example of plan adaptation, the robot discarded the held token to the trash because the number on the token was too large according to the current tokens on the board.
Event 63	An example of plan adaptation, the robot changed its target compartment because the human filled it.

Table 6.1: Collaborative and adaptive experiences stored in the validation NEEM.

new plan better suited to accomplish the goal (e.g. the robot went to another compartment when the human filled the one that the robot wanted to fill). Finally, the events showing non-collaborations (i.e. broken collaborations) represented cases when one of the axioms needed for a collaboration to exist was violated (e.g., the human stopped participating, or the goal/plan was not shared). Table 6.1 provides a description of each of the events.

6.4.3 Explanatory narratives generation: an example

The focus here is on one event among the twelve stored in the *validation NEEM*. *Event 15* shows the human stopping the collaboration (see Fig. 6.5). Using AXON with the parameters $\mathcal{G}_{\mathcal{T}} =$

(a) (b) (c)

Figure 6.5: Example of a non-collaboration in which the human stops participating in the shared task. (a) The human wears off its HTC tracker. (b) The human leaves the workspace. (c) The robot continues performing the task alone.

validation NEEM, $\mathcal{C} = Collaboration$, $(L_i, L_f) = (100.0, 142.0)$, $S = 3$, one obtains a narrative of the specific event. Recall that the level of specificity 3 includes the result of levels 1 (red) and 2 (blue).

'Event_15' (not) is a type of 'Collaboration' and is a type of 'Event' from 100.0 to 142.0 and executes plan 'Place Tokens In Columns By Color' and has participant 'Robot' and (not) has participant 'Human'. 'Place Tokens In Columns By Color' is a type of 'Plan' from 1.0 to 1000.0 and has component 'Full Board With Tokens In Columns By Color' and is plan of 'Robot and Human'. 'Robot' is a type of 'Physical Agent' from 1.0 to 1000.0 and has goal 'Full Board With Tokens In Columns By Color'. 'Human' is a type of 'Physical Agent' from 1.0 to 1000.0 and has goal 'Full Board With Tokens In Columns By Color'.

To see the rest of the generated narratives read Appendix D.

6.4.4 Pilot study: analysis of the usefulness of information

The length of an explanatory narrative plays a major role in the comprehension of its relevant information. The aim is to be informative, providing as much information as is needed, and no more [Grice, 1975]. Hence, a pilot study was carried out to assess the perceived usefulness of the narratives depending on their specificity, addressing RQ2.

Specifically, participants watched a video containing the twelve events included in the *validation NEEM*. The video depicted a textual narrative generated by our method after each of the events. Users were asked to imagine that they were about to receive training (the video with the narratives) aimed at preparing them to collaborate with a robot. This may be a real case in an industrial environment, where a video of a human-robot collaboration plus automatically generated narratives of the collaboration can be used to train new operators. A between-subject study was conducted, with three groups that evaluated each of the narratives' types. Groups 1, 2, and 3 evaluated the narratives with specificity 1, 2, and 3, respectively.

Procedure

The study was conducted in an isolated room to avoid distractions. The experimenter informed each participant of the procedure and asked them to fill out an informed consent form, in which they gave permission to gather their data for scientific purposes. Next, users were shown a warm-up video with the experiment's context and the narratives' format, ensuring that users received the same information before the experiment. Then, users watched the video with the twelve events recorded in the NEEM plus a textual narrative after each of the events. After watching the video, the participants were asked to fill out a questionnaire with two parts: information quality (usefulness) assessment, and open qualitative questions. The questionnaire is shown in Appendix C, and the videos are provided as supplemental material.[5]

Participants

30 participants (10 per group) were recruited. There was no withdrawal. Participants were aged between 21 and 59 (26.7% of them were female), with M=29 and SD=7.61. Most of them (93.3%) had a background in engineering, artificial intelligence, or robotics, and at least 70% had already interacted with other unspecified robots. Participation in the study was voluntary.

Quantitative and qualitative analysis

For a quantitative subjective analysis, it was used the quality of information measurement discussed by Lee et al. [Lee et al., 2002]. They presented a model for Information Quality, a questionnaire to measure it, and analysis techniques to interpret the measures. This chapter uses one of the quadrants of their model and its relative questionnaire: *usefulness*. It aims to assess whether or not the information is relevant to the user's task, in our case, the 'new operator training task'. In particular, *usefulness* was measured through five dimensions: *appropriate amount*, *relevancy*, *understandability*, *interpretability*, and *objectivity*. For each dimension, a set of questions had to be evaluated using an 11-point Likert scale ranging from completely disagree (0) to completely agree (10). The results of the study are shown in Fig. 6.6. Looking at them, one notes that the three levels of specificity produced useful narratives (all above 6.5 points). However, the second level (Group 2) was perceived as the most useful. Focusing on each dimension, the preferred narratives regarding the *appropriate amount* and the *interpretability* were those with specificity 2. Nevertheless, it is interesting to see that narratives with larger specificity (Group 3), were perceived to contain more *understandable* information. Other dimensions show negligible differences.

[5] www.iri.upc.edu/groups/perception/XONCRA

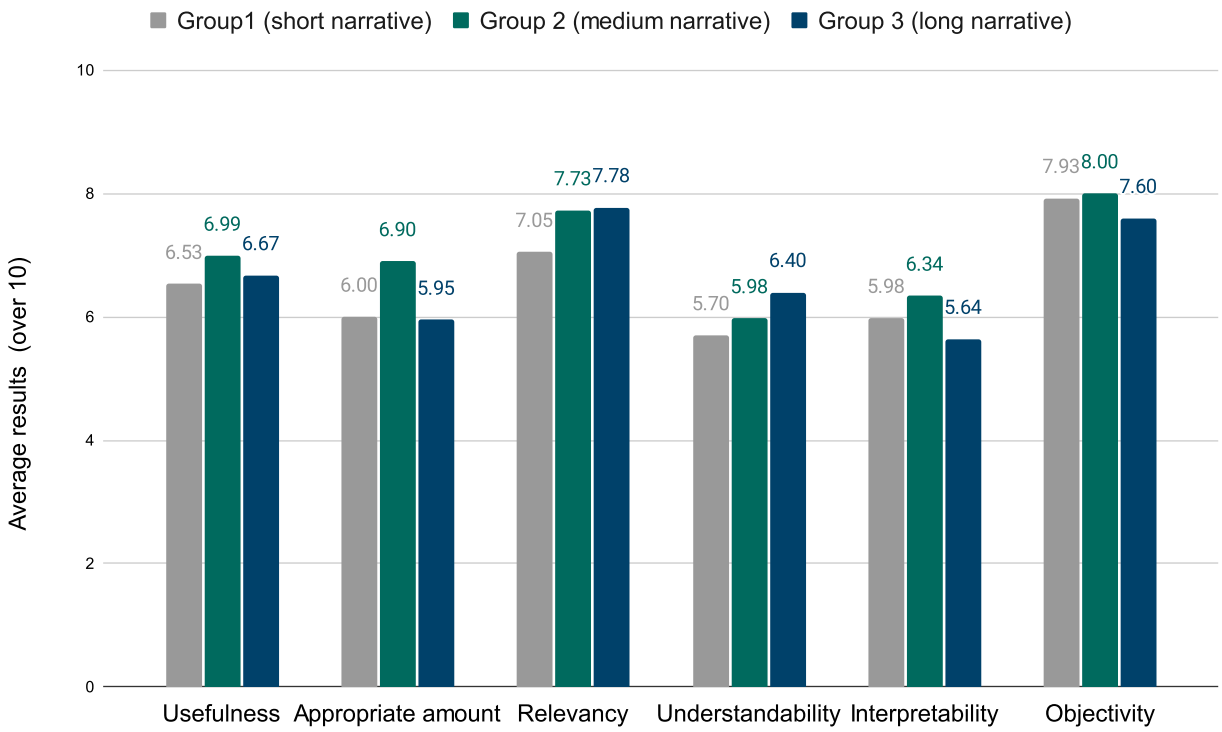


Figure 6.6: Results for the quantitative analysis of the information usefulness. Users' group number also corresponds to the specificity level of the assessed narratives. *Usefulness* is computed as the average of the other five dimensions.

The questionnaire also included some qualitative measures in the form of four open questions. First, the users were asked if a video without narratives would prepare them for real interaction with the robot and why. 86.7% of the participants answered that the video without explanation would not be enough to be prepared to collaborate with a real robot. This corroborated the need for a narrative, regardless of the specificity. The second question asked whether the explanation had helped to prepare them for real interaction with the robot and why. 66.7% found the narratives greatly helpful. However, this percentage changes if one looks at the isolated answer provided by each group: 50%, 70%, and 80% for Groups 1, 2, and 3, respectively. Hence, narratives with higher specificity seemed to be more helpful. Third, it was asked if they would prefer a summarized or a complete but repetitive narrative and why. 50% of the participants as a whole would prefer a summarized explanation. Nevertheless, that percentage grows to 70% for the participants of Group 3, who read longer narratives. Finally, it was asked if there was any content they would add to the narratives. Some participants proposed to include graphical information.

6.5 Discussion

This chapter introduced XONCRA, a methodology for the generation of explanatory ontology-based narratives for collaborative robotics and adaptation. It is built upon an existent ontology (OCRA) (see Chapter 5), and a knowledge-based episodic memory framework (NEEM) [Beetz et al., 2018]. These two elements together enable the representation, storage, and later retrieval of time-indexed knowledge. XONCRA also comprises a novel algorithm, AXON, which automatically retrieves knowledge from NEEMs to construct an explanatory narrative with it. It can produce three types of results based on the level of specificity. We provide an implementation of the methodology and some examples, addressing RQ1. The methodology sets an initial basis for ontology-based explainable robots, since it encompasses the three key elements of explainable robotics: representation, episodic memory, and explanation generation.

Depending on their specificity, the perceived narratives' usefulness was assessed through a pilot study, answering RQ2. Results indicated that participants found the three types to be useful. However, it was discovered that users preferred narratives generated with level 2 of specificity, especially for their appropriate amount and interpretability. Nevertheless, narratives with larger specificity (3), were perceived to contain more understandable information. The positive finding of this analysis is that all the narratives produced by XONCRA can help and be useful. Moreover, the methodology can address different preferences with respect to different trade-offs: appropriate amount vs understandability, etc.

Note that even though we focused on narratives of robot `Collaborations` and `Plan Adaptations`, the methodology generalizes beyond such a use case. By construction, it can deal with any other ontological class as long as it is formalized in the appropriate format to use the NEEMs framework. Indeed, there is a large list of available NEEMs generated for other purposes, e.g., a human setting up a table for breakfast, a robot monitoring a shelf in the retail domain, etc.[6] Utilizing those NEEMs, XONCRA might produce narratives about `Actions`, `Tasks`, `Objects`, etc. Indeed, we explore the use of the proposed approach to narrate pairs of alternative plans in Chapter 7.

We observe that thanks to the ontological representation of the information, one of the potential benefits of our proposal is that explanations can be adapted to the user and the situation. For example, starting by a shallow explanation and iteratively going deeper when more information is required. Or avoiding to repeat information when more than one explanation is required during the execution of a task. This is a very interesting research line

[6] https://neemgit.informatik.uni-bremen.de/neems

that needs to be explored in the future.

We have focused on leveraging the knowledge structure from the episodic memory to obtain the relevant information to form a proper explanation. One of the limitations of the presented approach is that the generated explanations are not natural enough, which may hinder their understanding. The improvement of our explanations using more sophisticated natural language techniques remains a research line to investigate in the future.

chapter seven

Beyond plain robot narratives: ontological contrastive explanations

> *..being able to embrace contradictions is a sign of intelligence..or insanity..*
>
> **— Richard Kadrey**
> (Butcher Bird)

Chapter 6 introduced the first ever ontology-based framework for explainable robots that comprises the three fundamental aspects of explainable agency: domain knowledge representation, an episodic memory, and explanation construction. The automatically generated explanations were satisfactorily evaluated with users, yielding promising results in fostering a shared understanding between humans and robots. The current chapter questions the scope of the proposed framework, challenging the coverage of the previously formalized knowledge, and investigating new types of explanations.

Hence, this work aims at generalizing the work presented in Chapters 5 and 6 beyond plain narratives of collaborative and adaptive experiences. First, through an ontological analysis, a new ontological model is obtained, augmenting the scope of the ontology proposed in Chapter 5. Specifically, a new theory for plan comparison is formalized, focusing on the properties and relationships that allow to compare plans. The robot's knowledge about the plans to compare is stored, and together with some logical rules, it is used to infer which plan is better. Second, a novel algorithm for contrastive explanatory ontology-based narratives is proposed, extending the methodology from Chapter 6 to contrastive explanation generation. From the robot knowledge, the algorithm retrieves the divergent information about the plans,

and then it constructs the final textual contrastive narrative. The proposed algorithm produces different types of narratives based on the chosen amount of detail (specificity), addressing different users' preferences. Based on objective evaluation metrics and using several planning domains, the algorithm is evaluated with respect to the original algorithm proposed in Chapter 6, which is used as a baseline. The proposed algorithm outperforms the baseline, using less knowledge to build the narratives (skipping repetitive knowledge), which shortens the time to communicate the narratives. Finally, it is briefly discussed how the proposed algorithm can be slightly modified to enhance and restrict the knowledge selection, which helps to shorten the constructed narratives and can be useful to personalize the explanations.

7.1 Motive

Autonomous artificial decision-making in environments with different agents (e.g., robots collaborating with or assisting humans) is complex to model. This is often due to the high degree of uncertainty and potential lack of communication among agents. For instance, robots might need to choose between competing plans, comparing their properties and deciding which one is better. Note that this decision-making problem is different from finding a single plan through automated planning, as here the idea is that there are already two valid plans to execute and the robot shall compare them and identify the best one. This might happen when a human gives an ambiguous command (e.g. 'can you bring me a drink?'), thus the robot may find different plans to achieve the abstract command (such as bringing any of the available drinks). Then it would be needed to compare and disambiguate the plans (see Figure 7.1). In these cases, mutual understanding of the ongoing decisions and communication between agents become crucial [Yuan et al., 2022]. Hence, trustworthy robots shall be able to model their plans' properties to make sound decisions when contrasting them. Furthermore, they shall also be capable of narrating (explaining) the knowledge acquired from the comparison. Note that robots add the possibility of physically executing the plan, which may affect the human, strongly motivating the need for explanations, which may serve two purposes: justifying the robot's selection of a plan, or asking the human to help in the disambiguation (i.e. the human may prefer the plan that the robot inferred as worse). Reflecting on these thoughts, this work addresses the following research questions:

- **RQ1 -** How could robots model and reason about what differentiates (two) plans, making one better?
- **RQ2 -** How could robots leverage the proposed ontological model to explain (narrate) what differentiates (two) plans?

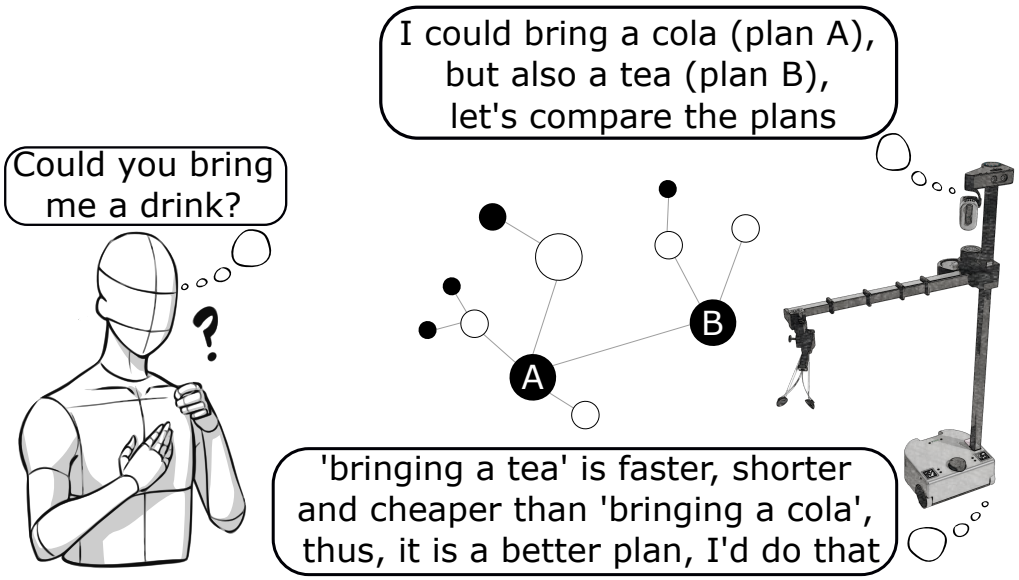


Figure 7.1: A prototypical scenario where a robot contrasts two plans and reasons about which plan is better to execute.

7.2 Related work

Concerning the modeling of domain knowledge for reasoning, a common approach is to use sound formalisms such as formal ontologies. The literature shows that multiple ontologies have been lately developed and even standardized for different robotic applications and domains [Schlenoff et al., 2012, Gonçalves et al., 2021b, Fiorini et al., 2017, Olivares-Alarcos et al., 2019a]. The literature also encompasses various works that have concentrated on ontologically modeling concepts related to plans and their properties. In this regard, Bermejo-Alonso [Bermejo-Alonso, 2018] conducted a survey of that domain, reviewing existing task and planning vocabularies, taxonomies, and ontologies, while also discussing their potential integration. The surveyed works mostly defined general concepts regarding plans, and only in some cases plan properties were discussed (e.g. cost and constraints).

A significant source of inspiration was discovered surrounding the notion of explainable agency (i.e., explaining the reasoning of goal-driven agents and robots). In their work, Langley et al. [Langley et al., 2017] advocated for the idea that explainable agency demands four distinct functional abilities. Among those, one can find the ability to explain decisions made during plan generation, involving the comparison of alternatives. Related to this, there are some literature efforts towards investigating how to boost explainable agency by narrating or verbalizing robots' internal knowledge about plans (e.g., a plan's sequence, the rationale to include a task in the plan, etc.) [Rosenthal et al., 2016, Flores et al., 2018, Canal et al.,

2022, Sridharan, 2023]. However, it is still unexplored how robots may explain their reasoning when comparing competing plans as a whole, not just specific plan tasks.

These two research domains, ontologies and explainable agency, are certainly connected. Langley et al. [Langley et al., 2017] also discussed that an explainable agent would need three main elements: a *representation* of the knowledge that supports explanations, an *episodic memory* to store agent experiences, and the ability to access the memory and retrieve and manipulate the stored content to *construct explanations*. Chapter 6 proposed a novel methodology comprising those three elements for the construction of explanatory ontology-based narratives for collaborative robotics and adaptation (XONCRA) [Olivares-Alarcos et al., 2023a]. It consisted of a knowledge base for collaborative robotics and adaptation (*know-cra*) that works as an episodic memory, and an algorithm to generate explanatory narratives using ontological knowledge (AXON). One might wonder whether the XONCRA methodology could be used to model and narrate the divergences between plans. Indeed, XONCRA uses the ontology OCRA [Olivares-Alarcos et al., 2022], presented in Chapter 5, which defined the relationship 'is better plan than', associating two plans denoting that one of the plans is considered to be better. However, it was not modeled in OCRA how an agent might find divergences between two plans and decide which one is better. Hence, it is necessary to propose a new ontological theory that formalizes the properties of plans for the purpose of contrasting them during robots' decision making. Moreover, while being a potential baseline solution, the narratives generated by AXON were not optimized for a contrastive case as the one that concerns this chapter, and a new approach shall be investigated.

7.3 Model for robot plan comparison

As stated before, there exist several useful methodologies to construct ontologies, e.g., [Fernández-López et al., 1997, Spyns et al., 2008], but none arise as a definite standard. Indeed, not all those methods are suitable for this work, in which the aim is to develop an ontological model from a foundational perspective (i.e. the characterization of the main concepts is more important than the coverage of the application domain). Hence, this chapter, just as Chapter 5 did, relies on ontological analysis, an approach that precedes the usual ontology construction process and aims to fix the core framework for the domain ontology. Based on this selection, the steps to perform are: to set the ontology domain and scope (competency questions), to enumerate, analyze and compare existing concepts (identification of shortcomings), to develop and formalize a more solid conceptualization, and to create

instances of the concepts and show their use (implementation/validation). Finally, it is also considered the documentation and maintenance of the produced theory.

7.3.1 Ontological scope of the proposed theory

The novel model will formalize the ontological classes and relationships to represent knowledge of plans and their characteristics for plan comparison and later contrastive narration. In order to scope the subject domain to be represented in the intended model, a set of competency questions is proposed, which are a set of requirements on the ontology content. Specifically, the proposed ontological model is expected to be able to answer the following questions:

- **CQ1 -** Which are the characteristics of a plan?
- **CQ2 -** How do the characteristics of different plans relate?
- **CQ3 -** How do different plans relate to each other?

The new model is going to be built upon OCRA, re-utilizing the existing model and extending it. Therefore, OCRA's upper ontology is inherited, the DOLCE+DnS Ultralite (DUL) foundational ontology [Borgo et al., 2021]. In addition to the proposed competency questions, for this work it is also interesting to represent the sequence of actions included in a plan, which is already covered by the DUL ontology.

7.3.2 Ontological shortcomings in OCRA and their theoretical remedy

Which are the characteristics of a plan? (CQ1)

OCRA did not define any ontological classes or relationships to model the properties of plans. Hence, an extension is required to be able to answer CQ1. For such an extension, a top-down approach is followed and the new model is built upon general entities defined in the upper-level ontology, DUL. In order to represent the features of other entities, DUL defines the class *Quality* as *'any aspect of an Entity (but not a part of it), which cannot exist without that Entity'*. DUL also includes the relationship *'has quality'*, *'a relation between entities and qualities'*. In this work, we specialize both the class and the relation to define the particular qualities of plans.

Plans can have many different qualities that would highly depend on the application domain. Defining all of them is out of the scope of this chapter. Instead, we aim to find a set of qualities that are usually present in most of the planning domains, with a special focus on those more relevant to robotics. Particularly, we will use temporal planning domains in which actions have a duration. In robotics, finding a valid plan is just the first part of the work to do, because

the focus is on the execution of the plan. Hence, considering the estimated duration (or makespan) of actions makes much more sense than in other artificial intelligence domains. After carefully studying temporal planning problems, it was discovered that three major generic qualities of plans were: cost, expected makespan, and number of tasks. Of course, one might consider as relevant other qualities (e.g., the expected risk of human injury, the probability of failure/success, the workload percentage among collaborative agents, etc.). We argue that our approach is easy to extend to accommodate the specific details of their applications. The proposed definition and formalization for each of the qualities is as follows:

Definition 7.1. *Plan Cost is a Quality that captures the cost of executing a Plan.*

$$\begin{aligned} PlanCost(q) \equiv dul.Quality(q) \ \wedge \\ \exists p \ \ dul.Plan(p) \wedge hasCost(p, q). \end{aligned} \tag{7.1}$$

Definition 7.2. *Plan Expected Makespan is a Quality that captures the expected time that would be required to execute a Plan.*

$$\begin{aligned} PlanExpectedMakespan(q) \equiv dul.Quality(q) \ \wedge \\ \exists p \ \ dul.Plan(p) \wedge hasExpectedMakespan(p, q). \end{aligned} \tag{7.2}$$

Definition 7.3. *Plan Number Of Tasks is a Quality that captures the number of tasks included in a Plan.*

$$\begin{aligned} PlanNumberOfTasks(q) \equiv dul.Quality(q) \ \wedge \\ \exists p \ \ dul.Plan(p) \wedge hasNumberOfTasks(p, q). \end{aligned} \tag{7.3}$$

The definitions include the notion of 'executing a plan', used here as a primitive which means *'following the sequence of tasks in the plan'*. The prefix *'dul'* denotes that a term was re-used from DUL, while novel terms and relations have no prefix. The plan's properties are modeled as qualities (in DUL) that are related to a plan they qualify. New relations between the plans and the qualities were introduced: *'has cost'*, *'has expected makespan'* and *'has number of tasks'*. Additionally, their inverse relations were also defined: *'is cost of'*, *'is expected makespan of'*, and *'is number of tasks of'*. These two pairs of three new relations were defined as specializations of the DUL's relations *'has quality'* and *'is quality of'*, respectively.

How do the characteristics of different plans relate? (CQ2)

The idea here is to be able to model knowledge such as: 'the characteristic Xa of plan Pa is worse than the characteristic Xb of Pb'. OCRA does not provide a formal way to compare the properties of plans. Considering the previously formalized classes, the aimed relations should hold between qualities of plans (e.g. *'PlanCost'*). Looking at the modeled qualities, one notices that all of them are numerical, hence, they could be related with comparative words such as: 'higher', 'lower', etc. However, it would be better to keep the new ontological model as reusable as possible, for instance, using more generic notions (e.g. worse/better quality). Indeed, qualities between plans can be worse and better, but also equal or equivalent, thus, this should also be modeled.

In total, three new transitive relations are formalized: *'is better quality than'*, *'is worse quality than'*, *'is equivalent quality to'*. The first two are inversely related, while the third one is symmetric. The three are defined as sub-relationships of the relation *'associated with'* (from DUL). They hold between two qualities, thus, they can be used beyond the scope of this work (e.g. comparing the qualities of robots, drinks, etc.). The proposed ontology is designed to model the outcome of comparing two qualities. However, it falls outside the scope of the model to make any commitment regarding the comparison criteria. Sec. 7.4.2 proposes some general inference rules for this. Users of the ontology may use them or define their own.

How do different plans relate to each other? (CQ3)

The OWL 2 DL version of OCRA defines the binary relationships *'is better plan than'* and *'is worse plan than'* which relate two plans stating that one of the plans is better or worse to achieve a goal. They were defined in the context of plan adaptations (i.e. events in which an agent decides to adapt an ongoing plan replacing it with a better option). Those two relations might be sufficient for the case of plan adaptations. Nevertheless, one might wonder what would happen in more general cases when two plans have equivalent properties. Neither of the plans would be better or worse than the other, thus, OCRA would fall short of modeling this. Furthermore, OCRA did not make any commitment about how an agent should compare plans and decide which one is better (see Sec. 7.4.2).

Given the lack of a formalization in OCRA on this matter, this chapter extends its coverage by formalizing specializations of the relationships *'is better plan than'* and *'is worse plan than'*. For each of them, three new sub-relations are introduced, one per quality: *'is cheaper plan than'*, *'is faster plan than'*, *'is shorter plan than'*; and *'is more expensive plan than'*, *'is slower plan than'*, *'is longer plan than'*, respectively. Furthermore, it is also added the relation *'is equivalent plan to'*, defined as disjoint with *'is better plan than'* and *'is worse plan than'*. Under this new relation, other three relations are created: *'is plan with same cost as'*, *'is plan with same expected makespan*

as', *'is plan with same number of tasks as'*. Recall that all these relations hold between two plans, answering CQ3.

7.3.3 Formalization of the model in OWL 2 DL

For practical use, the proposed ontological theory was formalized in OWL 2 DL. Hence, the axioms listed in this paper were translated to DL, more specifically, to the SROIQ(D) fragment of DL. It was possible to formulate each of the axioms in the target formalism with the exception of the value of plans' qualities. Note that quality is often used as a synonym for property but not in DUL, where qualities are particulars and properties are universals. In this regard, DUL considers that 'qualities inhere in entities' [Gangemi et al., 2003]. Every entity (including qualities) comes with its own exclusive qualities, which exist as long as the entity exists. DUL distinguishes between a quality (the cost of a plan) and its value or quale (a numerical data value). Hence, when saying that two plans have the same cost, their costs have the same quale, but still they are distinct qualities. This is convenient to model and answer CQ2, since OWL 2 DL cannot model relationships between two data values or quales (i.e. one cannot state that 5 is a better cost than 10). However, one can model the relation between the qualities (e.g. *'cost A'* has better quality value than *'cost B'*). Let's consider that a plan 'p' has a cost 'c' whose value is '10'. Hence, the knowledge would be modeled as:

$$PlanCost(c) \wedge dul.Plan(p) \wedge hasCost(p, c) \wedge hasDataValue(c, 10).$$

For consistency, the label of the relations comparing two qualities: *'is better quality than'*, *'is worse quality than'*, *'is equivalent quality to'*; were modified to *'has better quality value than'*, *'has worse quality value than'*, *'has equivalent quality value than'*.

7.3.4 Modeling the tasks of plans using DUL

The sequence of tasks described in a plan is one of the useful aspects of comparing plans. Therefore, modeling such knowledge would allow its use to narrate the differences between plans. The foundational ontology DUL already covers this knowledge, thus, there is no need for a new extension. As an example, let's imagine that there is a plan 'p' that consists in executing three tasks in the following order, 't1', 't2', and 't3'. This knowledge might be represented as follows:

$$\begin{gathered} dul.Task(t1) \wedge dul.Task(t2) \wedge dul.Task(t3) \ \wedge \\ dul.Plan(p) \wedge dul.definesTask(p, t1) \ \wedge \\ dul.definesTask(p, t2) \wedge dul.definesTask(p, t3) \ \wedge \\ dul.directlyFollows(t2, t1) \wedge dul.directlyFollows(t3, t2). \end{gathered}$$

Plan	*Nº of Tasks*	*Cost*	*Makespan (s)*
'bringing tea'	6	27	27
'bringing cola'	8	59	59

Table 7.1: Knowledge from the example of bringing drinks

7.4 The theory at work

The validation of an ontological model consists in creating instances of the concepts and showing their use. In this regard, the knowledge about instantiated plans (e.g., sequence of tasks, and qualities) would first be asserted to a knowledge base. Then, reasoning rules would be used to contrast the plans and infer which one is better. This section discusses the process of instantiating the model, and also introduces the reasoning rules used for contrasting the plans and their implementation. Finally, it shows how the model is able to answer the competency questions, validating the theory.

7.4.1 Instantiating the ontology with plans

The aim here is to demonstrate how to instantiate the model in a realistic scenario, providing as many resources as possible to potential users of the model. For this, the assertion of the knowledge about the plans was integrated with the planning system of the robot. Therefore, when a robot generates a plan by means of automated planning, it may also assert the knowledge about the plan, both its sequence and qualities. As a technical contribution, it was developed a novel knowledge-based framework that integrates existing robot planning tools and the use of the proposed ontology to model the comparison of plans (*know-plan*). This implementation is publicly available on a Github repository,[1] and illustrates how to instantiate the ontology with automatically generated plans. Planning is done using ROSPlan [Cashmore et al., 2015], a commonly used framework for planning in robotics. The ROSPlan framework provides a collection of tools for AI Planning for robots equipped with the Robot Operating System (ROS). Once a plan is found, the sequence of tasks and the plan's qualities are asserted to a knowledge base. As an example, a planning domain inspired by the scenario depicted in Fig. 7.1 is used, where a robot delivers drinks at an office. In order to have two plans to compare, the process is executed twice with different planning problems in which the drink that the robot should bring changes: a tea or a cola. A summary of the asserted knowledge of the plans' qualities is presented in Tab. 7.1.

[1] https://github.com/albertoOA/know_plan

7.4.2 Reasoning for plan comparison

The proposed ontological theory allows agents to represent the fact that a plan is better, worse or equally good than another one. However, none of the proposed axioms automates the comparison of plans and thus the inference of which plan is better.

Logical rules to infer the relation between plan's qualities

Let's assume that there is a consistent instantiated ontology $\mathcal{O}$ that contains knowledge about the qualities of different plans (P_a, P_b) as a set of triples $\langle subject, relation, object \rangle$ (see Sec. 7.4.1). The first step would be to compare the qualities' values and infer the relation between them (e.g. the cost of 'bringing cola' has a worse value than the cost of 'bringing tea'). Additionally, it can also be inferred the relation between the plans based on how the qualities relate (e.g. a plan is cheaper than another one). For instance, given the two plans (P_a, P_b), one can obtain their cost (C_a, C_b) and their cost' values (V_a, V_b) such that:

$$hasCost(P_a, C_a) \wedge hasCost(P_a, C_b) \wedge$$
$$dul.hasDataValue(C_a, V_a) \wedge dul.hasDataValue(C_b, V_b).$$

Then, if the values are equal ($V_a = V_b$) two new triples would be added to the knowledge base indicating that both costs have an equivalent quality value and that both plans have the same cost:

$$\mathcal{O} \leftarrow \mathcal{O} \cup \langle Ca, hasEquivalentQualityValueThan, Cb \rangle;$$
$$\mathcal{O} \leftarrow \mathcal{O} \cup \langle Pa, isPlanWithSameCostAs, Pb \rangle.$$

Similarly, when the values are different the asserted knowledge would refer to whether one of the costs/plans is worse/better than the other. Note that the cost's value is numerical, and it is usually assumed that the smaller it is, the better. Hence, when $V_a > V_b$:

$$\mathcal{O} \leftarrow \mathcal{O} \cup \langle Ca, hasWorseQualityValueThan, Cb \rangle;$$
$$\mathcal{O} \leftarrow \mathcal{O} \cup \langle Pa, isMoreExpensivePlanThan, Pb \rangle.$$

The final case would be when $V_a < V_b$, which would equally result in a triples' assertion of the inferred knowledge. As a whole, the complete described process becomes a logical rule to compare two plans' cost, which infers the relations between them and how this relation affects the connection between the plans. For the other qualities of plans formalized in the proposed model (expected makespan and number of tasks), analogous rules were defined. The criteria for the comparison were the same, since the rest of the qualities are numerical and for all of them, the lower their value is, the better.

Logical rule to infer which plan is better

Having inferred how the qualities and the plans relate, the next step is to infer which plan is better. Here, the criteria to decide if a plan is better than another one is satisfied when all the relations holding between them indicate that the plan has better qualities. This is, if all those relations are sub-properties of the relation *'is better plan than'*: $\forall R \langle Pa, R, Pb \rangle \rightarrow \langle R, rdfs.subpropertyOf, ocra.isBetterPlanThan \rangle$. The prefix 'rdfs' is used to denote that relations belong to the Resource Description Framework Schema (RDFS) [McBride, 2004]. When the condition is satisfied, the knowledge indicating that one plan is better would be asserted:

$$\mathcal{O} \leftarrow \mathcal{O} \cup \langle Pa, ocra.isBetterPlanThan, Pb \rangle.$$

It would similarly be defined for the cases of worse and equivalent plans, which as a whole would be the logical rule to infer, between two plans, which one is better. Note that this is general as depending on the application other criteria can be defined along with the corresponding rules. For instance, the effect of the different qualities might have a different weight (e.g. it may be more important to have a cheaper plan than a shorter one).

7.4.3 Implementation of the inference rules

Inherited from *know-cra*, this work uses Knowrob [Tenorth and Beetz, 2009, Beetz et al., 2018], a general framework for knowledge representation and reasoning for robots. The framework allows reading OWL-based ontologies and loading them into a knowledge base that is built using Prolog [Clocksin and Mellish, 2012]. Since the knowledge base is accessible through a prolog-based interface, it is possible to use the logical reasoning power of Prolog to make inferences. Therefore, the inference rules introduced in Sec. 7.4.2 can be implemented in Prolog. This would integrate the decision-making process to compare plans into the knowledge base, augmenting the reasoning capabilities of the proposed ontological model. Note that the implementation of *know-plan* also introduced some extra Prolog predicates to automate the call of the different rules. Specifically, it was implemented a predicate that runs all the rules for all the pairs of different plans stored in the knowledge base. The rules imply (binary) comparisons between pairs of qualities, and their complexity is linear with respect to the number of qualities, which would be added to the ontology (OWL 2 DL) complexity.

7.4.4 Answering the competency questions

The example of a robot delivering drinks (see Sec. 7.4.1) was used to showcase how to answer the competency questions. Note that the answers will contain the instantiated knowledge shown

in Tab. 7.1, plus inferred knowledge after applying the implemented rules. The queries are presented in prolog-like syntax (e.g. containing unbounded variables), since the knowledge base is written in Prolog.

Which are the characteristics of a plan? (CQ1)

$$triple('bringing\ \ tea', dul.hasQuality, Q), triple(Q, dul.hasDataValue, V).$$

If the query holds, i.e. the knowledge base contains the query triples, the answer contains an assignment of all the possible combinations of values of Q and V that make the query to be 'true'. Some examples of answers would be: 'cost of bringing tea' (Q) and '27' (V), or 'number of tasks of bringing tea' (Q) and '6' (V).

How do the characteristics of different plans relate? (CQ2)

$$triple('bringing\ \ tea', dul.hasQuality, Qa), triple('bringing\ \ cola', dul.hasQuality, Qb),$$
$$triple(Qa, R, Qb).$$

The answer would contain an assignment of all the possible combinations of values of Qa, R and Qb. For instance, 'cost of bringing tea' (Qa), 'has better quality value than' (R), and 'cost of bringing cola' (Qb). Note that this can only be answered after some of the inference rules have been applied (see Sec.7.4.2).

How do different plans relate to each other? (CQ3)

$$triple('bringing\ \ tea', R, 'bringing\ \ cola').$$

The answer would contain all the assignments to R that make the query to be 'true' in the knowledge base. In total, R could take four values: 'is cheaper plan than', 'is shorter plan than', 'is faster plan than', and 'is better plan than'. In order to answer this competency question, all the inference rules should have been applied.

7.5 Contrastive explanatory narratives of robot plans

7.5.1 May explanatory narratives do the work?

The ontological model proposed in this work augments the knowledge coverage in *know-cra* to model the divergences between plans (see Sec. 7.3), and to automate the inference of whether

those differences make one plan better than others (see Sec. 7.4). Hence, one might think that using the new model together with the XONCRA methodology from Chapter 6 would be enough to narrate what robots know about competing plans. In particular, given the knowledge about two plan instances and how they relate, XONCRA could produce a narrative about each of the plans using the AXON algorithm. The two narratives together would include the relevant knowledge for a robot to infer which plan is better, thus, humans could read the narratives and understand the inference. The differences between the two narratives could even be highlighted, as others have done when contrastively explaining the traces of two plans [Krarup et al., 2021]. However, such an approach would still require humans to extract their own conclusions by reading the complete narratives. Therefore, while being a potential baseline solution, the narratives generated by AXON do not seem to be optimized for the cases that concern this chapter, and a better approach might be developed.

7.5.2 Beyond plain explanatory narratives

Miller [Miller, 2019] stated that explanations are *contrastive*, *selected*, and *social*. Contrastive because they are sought in response to counterfactual cases that open questions such as: why a plan is better instead of others. Explanations are selected as they usually contain just part of the reasons, extracted by agents from a larger knowledge and based on specific criteria. Finally, explanations transfer knowledge in a conversational format, being part of a social interaction between agents. These three aspects of explanations set the basis to design a better algorithm, an alternative to AXON that:

- constructs contrastive narratives instead of plain narratives;
- enhances the selection of knowledge, extracting only the differences between the compared plans; and
- reduces the needed time to communicate a narrative, which might boost the (social) interaction.

7.5.3 Preliminary notation

Let's assume countable pairwise disjoint sets N_C, N_P, and N_I of class names, property names, and individuals, respectively. The standard relation *'rdf:type'*, which relates an individual with its class, is abbreviated as *'type'* and included in N_P. A knowledge graph $\mathcal{G}$ is a finite set of triples of the form $\langle s, p, o \rangle$ (subject, property, object), where $s \in N_I$, $p \in N_P$, $o \in N_I$ if $p \neq type$, and $o \in N_C$ otherwise. In this thesis, the knowledge base works as an episodic memory [Beetz et al.,

2018, Olivares-Alarcos et al., 2023a], thus it allows the assertion of triples with the time interval in which they hold. Hence, the stored knowledge can be seen as a time-indexed knowledge graph $\mathcal{G}_\mathcal{T}$, which is a finite set of tuples of the form $\langle s, p, o, t_i, t_f \rangle$, where $t_i, t_f \in \mathbb{R} > 0$, and denote the time interval (initial and final time) in which the triple $\langle s, p, o \rangle$ holds. Note that non-asserted knowledge is considered unknown and never false, since knowledge graphs usually comply with the open-world assumption. For this, the Web Ontology Language 2 (OWL 2) was developed to allow the explicit negative assertion of properties: $\langle s, p, o \rangle$ is $false$. Hence, $\mathcal{G}_\mathcal{T}$ may contain, e.g., that during an interval of time, $\langle t_i, t_f \rangle$, a task k is not defined in a plan p: $\langle k, dul : isDefinedIn, p, t_i, t_f \rangle$ is $false$. In this work, querying the $\mathcal{G}_\mathcal{T}$, we build what we called 'contrastive narrative tuples' of a pair of instance plans, $\mathcal{T}_\mathcal{P}$: $\langle s, p, o, t_i, t_f, sign \rangle$, where $sign$ indicates whether the time-indexed triple comes from a positive or negative assertion.

7.5.4 ACXON - An algorithm for contrastive explanatory ontology-based narratives

ACXON is a novel theoretical contribution, an algorithm that leverages the structure of the knowledge stored in episodic memories to retrieve knowledge about divergences between ontological entities (e.g. plans), for later construction of textual contrastive explanatory narratives. The algorithm takes as an input a time-indexed knowledge graph $\mathcal{G}_\mathcal{T}$ (as described in Sec. 7.5.3), the ontological class (or classes) of the pair of instances to narrate, the temporal locality (time interval of interest), and the level of specificity. Our focus is on contrastive narratives about *Plans*, but ACXON is general enough to work with other OWL 2 DL ontologies and classes. For instance, it might contrastively narrate the capabilities of a pair of agents (e.g. one can move for longer periods), or how two drinks are different to each other (e.g. one is healthier, tastier, etc.). Based on the level of specificity, there are three types of contrastive narratives. In this work, specificity refers to the amount of detail to be used during the narrative construction, i.e., the number of knowledge tuples.

ACXON (see Alg. 4) first retrieves a set $\mathcal{I}_{\mathcal{P}_\mathcal{T}}$ of sets comprising the instantiated pairs of the provided pair of classes $\mathcal{P}$, which exist, at least partially, within the temporal locality $\langle L_i, L_f \rangle$ (line 2). Each $\mathcal{I}_\mathcal{P}$ is a set of tuples $\langle e, t_i, t_f \rangle$, containing the two pair's instances e with their time interval (t_i, t_f) (line 4). Second, for each of the instantiated pairs $\langle \langle e_a, t_{i_a}, t_{f_a} \rangle, \langle e_b, t_{i_b}, t_{f_b} \rangle \rangle \in \mathcal{I}_{\mathcal{P}_\mathcal{T}}$, a set of knowledge tuples $\mathcal{T}_\mathcal{P}$ is retrieved according to the specificity level (line 5). Third, from the initial narrative tuples $\mathcal{T}_\mathcal{P}$, it is selected the sub-set containing only divergent knowledge between the pair of instances $\mathcal{D}_\mathcal{P}$ (line 6). Fourth, for each instantiated pair a contrastive explanation $\mathcal{E}_\mathcal{P}$ is built using their relative tuples (line 7). Finally, the new explanation is added to the set of explanations $\mathcal{E}$ (line 8). The implemented

Algorithm 4: ACXON

Input: Time-indexed knowledge graph ($\mathcal{G}_\mathcal{T}$), pairs to narrate ($\mathcal{P}$), temporal locality (L_i, L_f), specificity (S)
Output: Contrastive Narratives ($\mathcal{E}$)

$\mathcal{E} \leftarrow \emptyset$
$\mathcal{I}_{\mathcal{P}_\mathcal{T}} \leftarrow$ RetrieveInstantiatedPairsWithTime($\mathcal{G}_\mathcal{T}$, $\mathcal{P}, L_i, L_f$)
foreach $\langle\langle e_a, t_{i_a}, t_{f_a}\rangle, \langle e_b, t_{i_b}, t_{f_b}\rangle\rangle \in \mathcal{I}_{\mathcal{P}_\mathcal{T}}$ **do**
  $\mathcal{I}_\mathcal{P} \leftarrow \langle\langle e_a, t_{i_a}, t_{f_a}\rangle, \langle e_b, t_{i_b}, t_{f_b}\rangle\rangle$
  $\mathcal{T}_\mathcal{P} \leftarrow$ RetrieveNarrativeTuples($\mathcal{G}_\mathcal{T}, \mathcal{I}_\mathcal{P}, S$)
  $\mathcal{D}_\mathcal{P} \leftarrow$ ExtractDivergentNarrativeTuples($\mathcal{T}_\mathcal{P}$)
  $\mathcal{E}_\mathcal{P} \leftarrow$ ConstructContrastiveNarrative($\mathcal{D}_\mathcal{P}$)
  $\mathcal{E} \leftarrow \mathcal{E} \cup \mathcal{E}_\mathcal{P}$
end

algorithm and examples of its use are available online.[2]

Retrieve instantiated pair with time routine

With a time-indexed knowledge graph $\mathcal{G}_\mathcal{T}$, a pair of ontological existing classes or a set of them, $\mathcal{P} \subset N_C$, and the temporal locality $\langle L_i, L_f\rangle$, this routine retrieves a set of sets $\mathcal{I}_{\mathcal{P}_\mathcal{T}}$ comprising all the instantiated pairs of the given pair of classes, $\mathcal{I}_\mathcal{P}$, which contain the two time-indexed instances $\langle\langle e_a, t_{i_a}, t_{f_a}\rangle, \langle e_b, t_{i_b}, t_{f_b}\rangle\rangle$ of each pair such that their time intervals exist within (intersect) the temporal locality:

$$\begin{gathered}\forall\langle\langle e_a, t_{i_a}, t_{f_a}\rangle, \langle e_b, t_{i_b}, t_{f_b}\rangle\rangle \in \mathcal{I}_{\mathcal{P}_\mathcal{T}} \rightarrow \exists\langle c_a, c_b\rangle \in \mathcal{P} \wedge \\ \langle e_a, type, c_a, t_{i_a}, t_{f_a}, sign_a\rangle \in \mathcal{G}_\mathcal{T} \wedge (\langle t_{i_a}, t_{f_a}\rangle \cap \langle L_i, L_f\rangle) \wedge \\ \langle e_b, type, c_b, t_{i_b}, t_{f_b}, sign_b\rangle \in \mathcal{G}_\mathcal{T} \wedge (\langle t_{i_b}, t_{f_b}\rangle \cap \langle L_i, L_f\rangle)\ .\end{gathered}$$

Examples of time-indexed instances of plans to narrate may be:

$\langle\langle bringing\ \ water, 10.0, 150.0\rangle, \langle bringing\ \ juice, 0.0, 150.0\rangle\rangle$;
$\langle\langle bringing\ \ tea, \_\ , Inf\rangle, \langle bringing\ \ cola, \_\ , Inf\rangle\rangle$.

Note that the time interval is not always numerical, see the second example. This happens when a triple has held true in the knowledge base since an undetermined instant of time ('_'), and it will remain true as long as the knowledge base stays active ('Inf').

[2] www.iri.upc.edu/groups/perception/ontology-based-explainable-robots

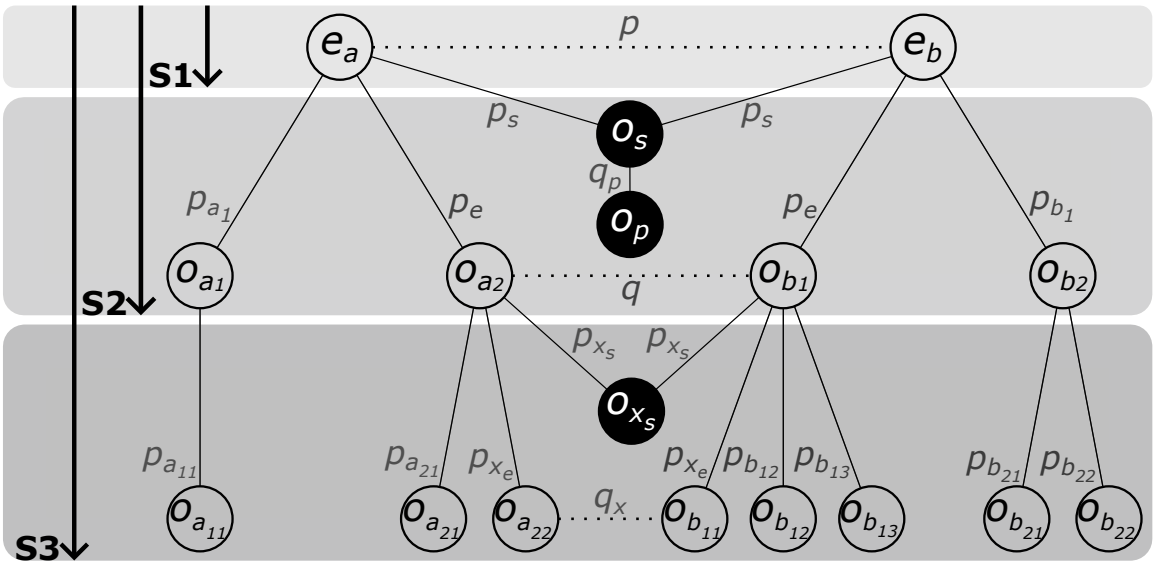


Figure 7.2: Graphical representation of the different levels of specificity (S1, S2, S3) and their respective depth in the knowledge graph. Nodes in black correspond to initially retrieved tuples that are later pruned because they are non-divergent.

Retrieve narrative tuples routine

For each instantiated pair to narrate $\mathcal{I}_\mathcal{P}$, with the $\mathcal{G}_\mathcal{T}$, and the level of specificity S, the routine would retrieve all the tuples about the pair, $\langle s, p, o, t_i, t_f, sign\rangle$, that are relevant to build the contrastive narrative. The first level of specificity retrieves tuples comprising all the relations p holding between the two instances of a pair: $\exists p \in N_P \wedge$ ($\langle e_a, p, e_b, t_i, t_f, sign\rangle \in \mathcal{G}_\mathcal{T} \vee \langle e_b, p, e_a, t_i, t_f, sign\rangle \in \mathcal{G}_\mathcal{T}$). In the second level, the routine adds all the tuples in which at least one of the instances $\langle e_a, e_b\rangle$ is related through a property p to an object o: $\exists p \in N_P \wedge$ ($\langle e_a, p, o, t_i, t_f, sign\rangle \in \mathcal{G}_\mathcal{T} \vee \langle e_b, p, o, t_i, t_f, sign\rangle \in \mathcal{G}_\mathcal{T}$). Following a similar logic to the first level, the second level also retrieves tuples that relate the different objects o. When the instances $\langle e_a, e_b\rangle$ are related to two objects $\langle o_a, o_b\rangle$ through the same property p, the tuples relating those two objects $\langle o_a, q, o_b, t_i, t_f, sign\rangle$ are also added to the list of narrative tuples: $\exists p, q \in N_P \wedge \langle o_a, q, o_b, t_i, t_f, sign\rangle \wedge \langle e_a, p, o_a, t_{i_a}, t_{f_a}, sign_a\rangle \in \mathcal{G}_\mathcal{T} \wedge \langle e_b, p, o_b, t_{i_b}, t_{f_b}, sign_b\rangle \in \mathcal{G}_\mathcal{T}$. These are horizontal links in Fig. 7.2.

Then, in the third level the routine adds all the tuples in which the objects o from the second level are related to other objects o_x: $\langle o, p_x, o_x, t_{ix}, t_{fx}, sign_x\rangle \in \mathcal{G}_\mathcal{T} \wedge \langle t_i, t_f\rangle \cap \langle t_{ix}, t_{fx}\rangle$. Robots' experiences may be tied to a time frame, thus, the search was restricted to tuples whose time interval $\langle t_{ix}, t_{fx}\rangle$ intersected the time interval of the pair's instances $\langle t_{i_a}, t_{f_a}\rangle$ and $\langle t_{i_b}, t_{f_b}\rangle$. This prevented the routine from retrieving tuples with irrelevant knowledge about the pair of instances to narrate $\langle e_a, e_b\rangle$. The third level finishes collecting the tuples relating the different

objects o_x between each other, similarly to how it is done in the second level: $\exists p_x, q_x \in N_P \wedge \langle o_{x_a}, q_x, o_{x_b}, t_i, t_f, sign\rangle \wedge \langle o_a, p_x, o_{x_a}, t_{i_a}, t_{f_a}, sign_{x_a}\rangle \in \mathcal{G}_\mathcal{T} \wedge \langle o_b, p_x, o_{x_b}, t_{i_b}, t_{f_b}, sign_{x_b}\rangle \in \mathcal{G}_\mathcal{T}$. For any of the levels, when the narrative's set of tuples $\mathcal{T}_\mathcal{P}$ already contains a tuple or its inverse, the tuple is not added. Furthermore, the tuples are retrieved incrementally from the first to the third level. Hence, when the specificity level is three, the returned tuples also contain those from the first and second levels. This would equate to moving deeper in the knowledge graph representing the instanced pair (see Fig. 7.2). Utilizing the ongoing example of bringing a drink (see Fig. 7.1), some instances of the retrieved narrative tuples $\mathcal{T}_\mathcal{P}$ of an instantiated pair are:

$\mathcal{T}_{\mathcal{P}_1}$ ⟨'bringing tea', isBetterPlanThan, 'bringing cola', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_2}$ ⟨'bringing tea', definesTask, 'T2-grasp object', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_3}$ ⟨'bringing tea', definesTask, 'task 0 - find person', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_4}$ ⟨'bringing cola', definesTask, 'task 0 - find person', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_5}$ ⟨'T3-go to waypoint', directlyPrecedes, 'T5-give object', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_6}$ ⟨'T7-give object', isTaskDefinedIn, 'bringing cola', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_7}$ ⟨'bringing tea', definesTask, 'T3-got to waypoint', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_8}$ ⟨'bringing tea', definesTask, 'T5-give object', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_9}$ ⟨'T3-go to waypoint', directlyFollows, 'T2-grasp object', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_{10}}$ ⟨'bringing tea', isCheaperPlanThan, 'bringing cola', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_{11}}$ ⟨'bringing cola', hasCost, 'cola cost', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_{12}}$ ⟨'bringing tea', hasCost, 'tea cost', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_{13}}$ ⟨'cola cost', hasDataValue, '59', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_{14}}$ ⟨'tea cost', hasDataValue, '27', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_{15}}$ ⟨'cola cost', hasWorseQualityValueThan, 'tea cost', _, Inf, positive⟩.

Extract divergent narrative tuples

From the initially selected narrative tuples $\mathcal{T}_\mathcal{P}$, the routine would just retrieve the set of knowledge tuples $\mathcal{D}_\mathcal{P}$ that capture divergences between the two pair's instances. The routine identifies the non-divergent tuples that exist in $\mathcal{T}_\mathcal{P}$ and prunes them. A pair of tuples $\langle\langle s_1, p_1, o_1, t_{i_1}, t_{f_1}, sign_1\rangle, \langle s_2, p_2, o_2, t_{i_2}, t_{f_2}, sign_2\rangle\rangle \in \mathcal{T}_\mathcal{P}$ will be non-divergent when: $(s_1 \neq s_2) \wedge (p_1 = p_2) \wedge (o_1 = o_2) \wedge (t_{i_1} = t_{i_2}) \wedge (t_{f_1} = t_{f_2}) \wedge (sign_1 = sign_2)$. Note that the routine prunes the whole branch of a non-divergent tuple, which includes tuples in which the shared object $(o_1 = o_2 = o_s)$ acts as the subject: $\langle o_s, q_p, o_p, t_i, t_f, sign\rangle$. The process will apply to tuples extracted at any of the specificity levels, as it is depicted in Fig. 7.2. In the example, after applying this routine, the tuples $\mathcal{T}_{\mathcal{P}3}$ and $\mathcal{T}_{\mathcal{P}4}$ would be pruned and the set of remaining tuples

would be:

$\mathcal{D}_{\mathcal{P}_1}$ ⟨'bringing tea', isBetterPlanThan, 'bringing cola', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}_2}$ ⟨'bringing tea', definesTask, 'T2-grasp object', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}_3}$ ⟨'T3-go to waypoint', directlyPrecedes, 'T5-give object', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}_4}$ ⟨'T7-give object', isTaskDefinedIn, 'bringing cola', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}_5}$ ⟨'bringing tea', definesTask, 'T3-got to waypoint', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}_6}$ ⟨'bringing tea', definesTask, 'T5-give object', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}_7}$ ⟨'T3-go to waypoint', directlyFollows, 'T2-grasp object', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}_8}$ ⟨'bringing tea', isCheaperPlanThan, 'bringing cola', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}_9}$ ⟨'bringing cola', hasCost, 'cola cost', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}_{10}}$ ⟨'bringing tea', hasCost, 'tea cost', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}_{11}}$ ⟨'cola cost', hasDataValue, '59', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}_{12}}$ ⟨'tea cost', hasDataValue, '27', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}_{13}}$ ⟨'cola cost', hasWorseQualityValueThan, 'tea cost', _, Inf, positive⟩.

Construct contrastive narrative routine

Considering the divergent tuples $\mathcal{D}_{\mathcal{P}}$ of a pair of instances $\langle e_a, e_b \rangle$ to narrate, the routine builds the contrastive explanatory narrative applying a set of rules: **casting**, **clustering**, **ordering**, and **grouping**. These rules describe aggregations commonly used by humans in natural language [Dalianis and Hovy, 1996].

Casting consists of homogenizing the ontological properties appearing in the tuples. First, by ensuring that for all the tuples $\mathcal{D}_{\mathcal{P}}$ that concern any of the pair's instances $\langle e_a, e_b \rangle$ the instances are the tuple's subject. In the ongoing example: $\mathcal{D}_{\mathcal{P}1}$, $\mathcal{D}_{\mathcal{P}2}$, $\mathcal{D}_{\mathcal{P}4}$, $\mathcal{D}_{\mathcal{P}5}$, $\mathcal{D}_{\mathcal{P}6}$, $\mathcal{D}_{\mathcal{P}8}$, $\mathcal{D}_{\mathcal{P}9}$, and $\mathcal{D}_{\mathcal{P}10}$. Hence, when $\mathcal{D}_{\mathcal{P}}$ contains a tuple in which any of the instances e acts as the object, $\langle s, p, e, t_i, t_f, sign \rangle \in \mathcal{D}_{\mathcal{P}}$, the casting rule reverses the tuple to: $\langle e, p^{-1}, s, t_i, t_f, sign \rangle$, where p^{-1} states for the inverse property of p. In the list of tuples from before, the tuple $\mathcal{D}_{\mathcal{P}4}$ that contains the property *'isTaskDefinedIn'* would be reversed using *'definesTask'*. Then, all the tuples concerning any of the two instances, both reversed tuples and those that did not need to be inverted, are added to a new set $\mathcal{D}_{\mathcal{P}Cast}$ of cast tuples. Casting has a second step that involves the tuples not concerning the pair's instances ($\mathcal{D}_{\mathcal{P}3}$, $\mathcal{D}_{\mathcal{P}7}$, $\mathcal{D}_{\mathcal{P}11}$, $\mathcal{D}_{\mathcal{P}12}$, and $\mathcal{D}_{\mathcal{P}13}$). Guaranteeing that the properties of the tuples to add are consistent with those that already exist in the cast tuples. Hence, if this is not the case, the tuple is reversed before adding it to $\mathcal{D}_{\mathcal{P}Cast}$. In the example, $\mathcal{D}_{\mathcal{P}3}$ is added to $\mathcal{D}_{\mathcal{P}Cast}$ (following the order) thus, $\mathcal{D}_{\mathcal{P}7}$ needs to be inverted before added. $\mathcal{D}_{\mathcal{P}11}$, $\mathcal{D}_{\mathcal{P}12}$, $\mathcal{D}_{\mathcal{P}13}$ are just added.

Next, this routine applies **clustering**, which structures the tuples in a way that each cluster will later be used to form a single sentence within the whole narrative. The narratives shall contrast the knowledge relating the pair's instances, revealing the divergences between them. Therefore, an effective strategy for cluster generation is to utilize the structure of the knowledge graph formed by the retrieved tuples (see Fig. 7.2). First, a cluster is created with the tuples $\langle e_a, p, e_b, t_i, t_f, sign\rangle$ that relate the pair's instances $\langle e_a, e_b\rangle$, in the example, $\mathcal{D}_{\mathcal{P}1}$ and $\mathcal{D}_{\mathcal{P}8}$. Second, the routine clusters the remaining tuples by property p in three different steps named: *direct*, *indirect*, *unrelated*. The tuples *directly* related through p to the instances to compare: $\langle e_a, p, o_a, t_i, t_f, sign\rangle$, $\langle e_b, p, o_b, t_i, t_f, sign\rangle$, are clustered together with the tuples relating their objects: $\langle o_a, q, e_b, t_i, t_f, sign\rangle$. Note that when only one of the instances $\langle e_a, e_b\rangle$ is related to an object through p, e.g., $\langle e_a, p, o_a, t_i, t_f, sign\rangle \in \mathcal{D}_{\mathcal{P}} \wedge \langle e_b, p, o_b, t_i, t_f, sign\rangle \notin \mathcal{D}_{\mathcal{P}}$; the knowledge will be clustered later at the *unrelated* step. In the example, $\mathcal{D}_{\mathcal{P}9}$, $\mathcal{D}_{\mathcal{P}10}$, $\mathcal{D}_{\mathcal{P}13}$ form a cluster, and $\mathcal{D}_{\mathcal{P}2}$, $\mathcal{D}_{\mathcal{P}5}$, $\mathcal{D}_{\mathcal{P}6}$ and the reversed $\mathcal{D}_{\mathcal{P}4}$ another one. New clusters are created for the tuples *indirectly* related to the instances to compare, i.e. those related to the objects $\langle o_a, o_b\rangle$ of the previous step: $\langle o_a, p_x, o_{x_a}, t_{i_a}, t_{f_a}, sign_{x_a}\rangle$, $\langle o_b, p_x, o_{x_b}, t_{i_b}, t_{f_b}, sign_{x_b}\rangle$. As before, those clusters include the tuples holding between their objects $\langle o_{x_a}, o_{x_b}\rangle$. Recall that the tuples are only clustered if the objects $\langle o_a, o_b\rangle$ are each related to one of the main instances to compare. In the example, $\mathcal{D}_{\mathcal{P}11}$ and $\mathcal{D}_{\mathcal{P}12}$ form a cluster. Finally, (*unrelated*) clusters are created with the remaining tuples sharing the same property p, thus $\mathcal{D}_{\mathcal{P}3}$ and the inverted $\mathcal{D}_{\mathcal{P}7}$ are clustered.

Subsequently, the clustered tuples are **ordered** externally (between clusters) and internally (between tuples). When externally ordering, the set of clusters is ordered according to the sequence followed during the clustering: first the cluster relating the pair's instances followed by the clusters obtained in the *direct*, *indirect*, and *unrelated* steps. Then, within the clusters from a single step, the clusters are ordered from more knowledge (more tuples) to less. The internal ordering just assures that the tuples with the property $p = type$ are at the front of each of the clusters. In the example, after applying all these rules the set of tuples would now be:

$\mathcal{D}_{\mathcal{P}1}$ ⟨'bringing tea', isBetterPlanThan, 'bringing cola', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}2}$ ⟨'bringing tea', isCheaperPlanThan, 'bringing cola', _, Inf, positive⟩,

$\mathcal{D}_{\mathcal{P}3}$ ⟨'bringing tea', definesTask, 'T2-grasp object', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}4}$ ⟨'bringing tea', definesTask, 'T3-got to waypoint', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}5}$ ⟨'bringing tea', definesTask, 'T5-give object', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}6}$ ⟨'bringing cola', definesTask, 'T7-give object', _, Inf, positive⟩,

$\mathcal{D}_{\mathcal{P}7}$ ⟨'bringing cola', hasCost, 'cola cost', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}8}$ ⟨'bringing tea', hasCost, 'tea cost', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}9}$ ⟨'cola cost', hasWorseQualityValueThan, 'tea cost', _, Inf, positive⟩,

$\mathcal{D}_{\mathcal{P}10}$ ⟨'cola cost', hasDataValue, '59', _, Inf, positive⟩,
$\mathcal{D}_{\mathcal{P}11}$ ⟨'tea cost', hasDataValue, '27', _, Inf, positive⟩,

$\mathcal{D}_{\mathcal{P}12}$ ⟨'T3-go to waypoint', directlyPrecedes, 'T5-give object', _, Inf, pos.⟩,
$\mathcal{D}_{\mathcal{P}13}$ ⟨'T2-grasp object', directlyPrecedes, 'T3-go to waypoint', _, Inf, pos.⟩.

Finally, the tuples of each cluster pass through several **grouping** steps, obtaining the sentences of the final textual contrastive narrative $\mathcal{E}_{\mathcal{P}}$. First, object grouping, the tuples sharing subject, property, interval, and sign are united. Thus, given $\langle s, p, o_a, t_i, t_f, sign\rangle$ and $\langle s, p, o_b, t_i, t_f, sign\rangle$, at this step the routine unites them to: $\langle s, p, o_a\ and\ o_b, t_i, t_f, sign\rangle$. In the example, $\mathcal{D}_{\mathcal{P}3}$, $\mathcal{D}_{\mathcal{P}4}$, and $\mathcal{D}_{\mathcal{P}5}$ are grouped into: ⟨'bringing tea', definesTask, 'T2 - grasp object' and 'T3 - got to waypoint' and 'T5 - give object', _, Inf, positive⟩. The second step consists in grouping tuples by predicate (property), thus tuples sharing subject, object, interval, and sign are joined. Considering two tuples: $\langle s, p_a, o, t_i, t_f, sign\rangle$ and $\langle s, p_b, o, t_i, t_f, sign\rangle$, the routine unites them to: $\langle s, p_a\ and\ p_b, o, t_i, t_f, sign\rangle$. In the example, $\mathcal{D}_{\mathcal{P}1}$ and $\mathcal{D}_{\mathcal{P}2}$ would be grouped. The final grouping stage generates textual contrastive sentences for the knowledge stored in each of the clusters of tuples by translating the tuples into text and connecting them. For instance, a cluster may contain the tuples $\langle s, p, o, t_i, t_f, sign\rangle$ and $\langle o, q, o_x, t_{i_x}, t_{f_x}, sign\rangle$. Hence, the knowledge is connected as a subordinate sentence using the pronoun 'which'. In the example, this happens with $\mathcal{D}_{\mathcal{P}7}$ and $\mathcal{D}_{\mathcal{P}9}$. Note that when the subordinate is introduced in tuples that have gone through object grouping, it is also added the phrase 'and also' for readability purposes. The explanations are contrastive, thus the conjunction 'while' is added to emphasize the divergent (contrastive) knowledge: e.g. $\langle e_a, p, o_a, t_{i_a}, t_{f_a}, sign_a\rangle$ and $\langle e_b, p, o_b, t_{i_b}, t_{f_b}, sign_b\rangle$. In the example, 'while' is used to compare the knowledge from the grouped $\mathcal{D}_{\mathcal{P}7}$ and $\mathcal{D}_{\mathcal{P}9}$, and $\mathcal{D}_{\mathcal{P}8}$; and $\mathcal{D}_{\mathcal{P}6}$ and the grouped $\mathcal{D}_{\mathcal{P}3}$, $\mathcal{D}_{\mathcal{P}4}$, $\mathcal{D}_{\mathcal{P}5}$. The the same applies for *indirectly* related clusters such as the one including tuples $\mathcal{D}_{\mathcal{P}10}$ and $\mathcal{D}_{\mathcal{P}11}$. The tuples from *unrelated* clusters, e.g. $\mathcal{D}_{\mathcal{P}12}$ and $\mathcal{D}_{\mathcal{P}13}$, are connected using 'and'. The propositions

'from' and 'to' are also added to introduce the tuples' time intervals, but only if they are different to the interval of the pair's instances. Indeed, if the interval is undetermined (_, Inf), it is obviated. The names of ontological entities (instances, classes and properties) are kept, only some properties are slightly changed to more understandable terms (e.g. using 'includes task' instead of 'definesTask', or 'has a higher value than' instead of 'hasWorseQualityValueThan'). The final narrative for the ongoing example would be:

> 'bringing tea' is better plan than and is cheaper plan than 'bringing cola'. 'bringing tea' includes task 'T2-grasp object' and 'T3-got to waypoint' and 'T5-give object', while 'bringing cola' includes task 'T7-give object'. 'bringing cola' has cost 'cola cost', which has a higher value than 'tea cost'; while 'bringing tea' has cost 'tea cost'. 'cola cost' has value '59', while 'tea cost' has value '27'. 'T3-go to waypoint' directly precedes 'T5-give object', and 'T2-grasp object' directly precedes 'T3-go to waypoint'.

7.6 Evaluating explanatory narratives

To evaluate the quality of the narratives generated by ACXON, the AXON algorithm from Chapter 6 was used as a baseline. Specifically, both algorithms were used to narrate the knowledge about contrasting plans. Following the ideas discussed in Sec. 7.5.1, these two algorithms can be compared by using AXON twice (i.e. to narrate each of the plans independently). By construction, ACXON is expected to reduce the amount of knowledge used in the explanations, and also to ensure shorter explanation communication times.

7.6.1 Evaluation procedure and setup

The evaluation was done using a set of temporal planning PDDL domains from recent international planning competitions (IPC) [Fox and Long, 2003]. The set included the IPC'02 Rovers domain [Fox and Long, 2003] (10 instantiated problems), the IPC'08 Crew planning domain [Barreiro et al., 2009] (15 problems), and the IPC'14 Match cellar domain [Halsey et al., 2004] (20 problems). First, given a domain and two problems, we run a planner with both problems to obtain two plans for which their respective knowledge (sequence and qualities) is instantiated as described in Sec. 7.4.1. Then, the inference rules are applied, performing the comparison of the two plans and asserting the inferred knowledge (e.g. which plan is better). Finally, both algorithms are used to extract the knowledge from the active knowledge base and construct the explanations. The algorithms have the same inputs: a time-indexed graph (the active knowledge base), a time interval (_, Inf) and the level of specificity (the three levels were used). A set of metrics discussed in Sec. 7.6.2 are computed

for each of the generated narratives. Note that for each of the planning domains, five pairs of instances were randomly selected (fifteen in total). The software for the test was run on a desktop PC with an Intel Core i7-8700K CPU (12x 3.70 GHz), 16 GB DDR4 RAM, and an NVIDIA GeForce GT 710/PCIe/SSE2 GPU.

7.6.2 Metrics for explanation evaluation

To evaluate the explanations we have selected a set of offline objective evaluation metrics, aligned with the existing literature. The metrics aim to evaluate two of the main features of explanations: the selection of content (number of attributes), and the social aspect (communication time and readability).

Number of attributes. The metric is commonly used to evaluate explainable models. Especially when evaluating the explainability of black box models (e.g. machine learning (ML) models) [Rosenfeld, 2021]. Nevertheless, this metric has also been used to evaluate non-ML explanatory systems [Georgara et al., 2022]. In this work, the number of attributes is equal to the number of tuples $\mathcal{D}_{\mathcal{P}}$ used to construct the narratives.

Communication time. Explanations are social, thus a good quality index is to measure how much time would require an agent to communicate them. In this work, the communication time is computed as a combination of the *construction time* C_T and the actual *interaction time* I_T. For the interaction, two channels are considered: auditory (I_{TA}) and visual (I_{TV}). For retaining information, people are comfortable with a speaking pace of 150-160 words per minute (wpm), while the pace for silent reading is 250-400 wpm [Rayner et al., 2016]. In this work, the interaction time is estimated by counting the number of words in the narratives and considering the fastest pace for each channel: 160 wpm (auditory), and 400 wpm (visual).

Readability metric. Since the narratives are generated using natural language, the Dale–Chall readability R_{DC} formula [Chall and Dale, 1995], a well-known readability metric, is also used. Most of the readability metrics use a similar formula including two terms: (a) the proportion of 'complex words' relative to the total number of words; and (b) the number of words per sentence. Usually, the word length or a number of syllables is used to decide whether a text's words are 'complex' (i.e. difficult to understand). More interestingly, Dale-Chall defines words as 'complex' if they are not familiar (i.e. not included in a list of 3000 common words) [Chall and Dale, 1995]. The resulting score indicates the reading level by educational grade needed to comprehend the text.

7.6.3 Results of the evaluation and discussion

The average evaluation results for the fifteen pairs of plans and each algorithm and level of specificity are summarized in Tab. 7.2. ACXON outperforms the baseline method in most cases, especially for levels two and three of specificity. Regarding the **number of tuples** $\mathcal{D}_{\mathcal{P}}$, using ACXON results in an overall reduction of more than 40% and 70% for levels 2 and 3 respectively. This is because ACXON does a better selection of the narrative tuples. First, by avoiding repeated tuples by collecting them for the whole pair instead of individually selecting tuples for each of the instantiated plans (*retrieve narrative tuples* routine in Sec. 7.5.4). Second, by pruning the non-divergent knowledge between the plans (*extract divergent narrative tuples* routine in Sec. 7.5.4). Hence, the contrastive narratives only contain what makes the plans different without undesired repetitions. For the **construction time** C_T, there are no major differences. However, it is worth commenting that the generation for level 3 would even require more than two seconds on average for any of the algorithms. Note that in some cases, the narrated plans contained more than 50 actions, hence, narratives of level 3 were long. ACXON produces a decrease in the **interaction times** I_{TA} and I_{TV} of approximately the 40% and 70% for specificity 2 and 3, respectively. The average times for the baseline method would be completely prohibitive for realistic interaction with humans. For level 3 of specificity, it would take 13 and 5 minutes for a human to listen and read the narratives, respectively. Indeed, although ACXON reduces those times to 4 and 1.5 minutes, the improvement still falls short of ensuring a fluent and socially acceptable interaction. To overcome this, the robot might provide the short explanation of level 1, and only more details if required. Concerning this, ACXON produces longer times when the specificity is 1. This is because ACXON is more informative than the baseline method since it includes the relationships between the plans at that level (e.g. shorter, better plan, etc.). Meanwhile, the baseline algorithm only states that both plans are instances of the class *'Plan'*, refer to Chapter 6 for more details about the baseline. Finally, in relation to the **readability index** R_{DC} both behave similarly with a metric value close to 9, denoting a high portion of complex words. Specifically, such a value indicates that the explanations would be easily understood by an average college student. Hence, people with a lower education level may require a higher effort to interpret the narratives. Interestingly, the baseline method obtains a better value for specificity 3, because its narratives contain multiple short sentences, which is favored in the metric. In ACXON, since the narratives are contrastive, they contain several subordinate sentences, which results in a higher degree of complexity.

Method	**Baseline** (AXON)			**ACXON**		
Specificity	1	2	3	1	2	3
$\mathcal{D}_{\mathcal{P}}$	**2.00**	77.40	305.73	3.80	**44.27**	**91.33**
C_T *(s)*	0.73	0.87	3.50	**0.56**	**0.75**	**2.53**
I_{TA} *(s)*	**5.25**	170.55	782.80	8.68	**101.00**	**258.85**
I_{TV} *(s)*	**2.10**	68.22	313.12	3.47	**40.40**	**103.54**
R_{DC}	9.62	**8.86**	**2.81**	**8.78**	9.24	7.88

Table 7.2: Average evaluation results for the 15 pairs of plans.

7.7 What if explanations were more selective?

The evaluation results demonstrated that ACXON enhances the selection of knowledge for explanatory contrastive narratives with respect to the baseline. However, it still requires long communication times for specificity levels 2 and 3, thus it shall be more selective. For instance, one might use the structure of the knowledge to constrain the tuples retrieval performed by the *retrieve narrative tuples* routine from Sec. 7.5.4, focusing on part of the contrastive knowledge (e.g. only the plans' qualities).

Following this rationale, it is proposed here a modification to the *retrieve narrative tuples* routine to focus on specific aspects of plans. Specifically, at the second level of specificity, when the routine adds all the tuples in which at least one of the instances $\langle e_a, e_b \rangle$ is related through a property p to an object o. Instead of considering tuples containing any object, the ontological class c of the object is restricted: $\exists p \in N_P \wedge \exists c \in N_C \wedge (\langle e_a, p, o, t_i, t_f, sign \rangle \in \mathcal{G}_{\mathcal{T}} \vee \langle e_b, p, o, t_i, t_f, sign \rangle \in \mathcal{G}_{\mathcal{T}}) \wedge \langle o, type, c, t_i, t_f, sign \rangle$. This minor modification reduces the number of tuples retrieved at level 2, but its effect is also propagated to level 3, producing a larger decrease in the final number. Let's imagine that in the ongoing example from before, the algorithm is asked to compare the plans only using the qualities of plans (i.e. instances of *dul.Quality*), the retrieved tuples would be reduced from 15 to 7:

$\mathcal{T}_{\mathcal{P}_1}$ ⟨'bringing tea', isBetterPlanThan, 'bringing cola', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_2}$ ⟨'bringing tea', isCheaperPlanThan, 'bringing cola', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_3}$ ⟨'bringing cola', hasCost, 'cola cost', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_4}$ ⟨'bringing tea', hasCost, 'tea cost', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_5}$ ⟨'cola cost', hasDataValue, '59', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_6}$ ⟨'tea cost', hasDataValue, '27', _, Inf, positive⟩,
$\mathcal{T}_{\mathcal{P}_7}$ ⟨'cola cost', hasWorseQualityValueThan, 'tea cost', _, Inf, positive⟩.

The rest of the algorithm's routines would be applied as was shown before, producing a shorter narrative:

‘bringing tea’ is better plan than and is cheaper plan than ‘bringing cola’. ‘bringing cola’ has cost ‘cola cost’, which has a higher value than ‘tea cost’; while ‘bringing tea’ has cost ‘tea cost’. ‘cola cost’ has value ‘59’, while ‘tea cost’ has value ‘27’.

With this modification ACXON does a better selection of the knowledge used in the contrastive narrative, shortening the explanations and reducing the communication time. Furthermore, the algorithm now captures the preferred content to use in contrastive narratives, which might be used to provide personalized explanations. Hence, the modification, while being simple, contributes to generating potentially more socially acceptable explanations.

7.8 Discussion

This work presents a method for robots to model and reason about the differences between plans, to infer which one is better and to narrate the inferences to other agents (e.g. humans). The approach comprises a novel ontological model for robots to describe plans and their qualities for reasoning during plans comparison, and a new algorithm to construct ontology-based contrastive explanatory narratives. The approach is general to be used with other ontologies, beyond the case of contrasting plans (e.g. modeling the qualities of two drinks and narrating the differences). The model is validated by instantiating it to answer a set of competency questions. The algorithm is evaluated against a baseline with respect to a set of objective metrics. Our solution outperforms the baseline in general, doing a better selection of knowledge tuples to build the explanation (avoiding non-divergent knowledge), and producing explanations that would require less time for the robot to communicate them. In the future, we will look at making more accessible the narratives’ language, and a user study will be conducted to evaluate their quality. Note that robots add the possibility of physically executing the plan, which opens issues to investigate: the ‘preferred’ moment to explain (e.g. before or after executing), or the context in which the competing plans are conceived (e.g. due to an adaptation while executing).

chapter eight

Conclusion

> *..¡qué imprudencia más ridícula hablar de lo definitivo! ¡qué ganas de cerrar puertas que han de abrir los que vengan detrás!..*
>
> *..dejemos las conclusiones para los imbéciles..*
>
> **— Pío Baroja**
> (*La ruta del aventurero*)

This thesis demonstrates the viability of employing ontologies as an integrative framework for constructing robot explanations, particularly within interactive settings involving humans. Each chapter navigates the reader through an exhaustive research expedition culminating in the establishment of a solid foundational basis for ontology-based explainable robots. The expedition commences by exploring the literature on ontological frameworks to support robot autonomy (Chapter 2), helping us to do an informed selection of the target reality phenomena to be conceptualized. The exploration continues by acquiring insightful hands-on experience developing novel robot perception methods (Chapters 3 and 4), which served to grasp a proper understanding on the domain knowledge and frame the ontological scope of the models proposed in the thesis. Building upon insights acquired in these initial stages, the journey progresses into ontological conceptual modeling (Chapters 5 and 7), and it finally arrives to the utilization of these models for fostering explainable agency in robotics (Chapters 6 and 7).

8.1 Findings and lessons learned

From the exploration of the state of the art, we understood that a proper literature review is of especial relevance in the research domain where applied ontology and robotics intersect. In this domain, a robotics engineer might be tempted to engineering a new ontology for a specific robotic task without making the effort of finding an existing model and reusing it. Such a practice would indeed be contrary to the essence of applied ontology (i.e. understanding, clarifying, making explicit and communicating people's assumptions about the nature and structure of the world). When analyzing and reviewing different works, we found similar concepts defined in multiple ontologies at the same time. This is perfectly understandable, since the conceptualization process might be biased by researchers' background and needs. However, the existence of multiple inconsistent definitions that do not properly acknowledge each other hinders research advancements in the domain. Hence, new research works shall identify potential conflicts between existing definitions, and propose a novel conceptualization when there is a justified need for it. From experience gained in Chapters 5 and 7, we know that those issues can be alleviated and even prevented by relying on ontological analysis, an approach that precedes the usual ontology construction process and aims to fix the core framework for the domain ontology.

Ontological analysis is especially useful when developing an ontology from a foundational viewpoint where the characterization of the core concepts is more important than the coverage of the application domain. A foundational perspective helps with the construction of a flexible and general ontology, often small, which can be applied or easily adapted to different domains and applications, thus facilitating the ontology's reusability. Indeed, this perspective is often considered when developing upper-level (i.e. general) ontologies. However, one might wonder if such a general approach is appropriate for the robotics domain. General and reusable concepts might be of little help for actual reasoning tasks needed in a realistic robotic application. Hence, there is indeed a trade-off between flexibility and applicability, which is especially prominent in robotics due to its practical nature. In this thesis, such a trade-off is successfully addressed by starting the ontological analysis from a deep understanding of the target robotic applications, which was gained from the hands-on experience developing robot perception tasks (Chapters 3 and 4).

Related to the idea of starting from actual robotic tasks, it is easy to face a challenge that we here refer to as *the relevance problem*. Robots are often exposed to huge amounts of internal and external data, and it can be difficult to find the relevant data to abstract into knowledge (i.e. the relevant knowledge to conceptualize). An apprentice ontology developer may opt for abstracting as much data as possible, constructing large ontologies that would hopefully allow

robots to perform many complex reasoning tasks. However, the process in which the perception data is abstracted may add some complexity to the robotic system, presenting some scalability issues. Indeed, abstracting all the data the robot is exposed to would certainly result in codifying huge volumes of seemingly irrelevant knowledge. In order to mitigate these issues, it was useful to look at this problem from a foundational viewpoint. Specifically, we carefully identified the knowledge to conceptualize considering the robot reasoning tasks that would make sense to solve using ontological knowledge. From the cognitive capabilities considered in Chapter 2, we focused on *recognition and categorization*, and *decision making and choice*, barely covered in the literature. Hence, we set the perception tasks with the aim to recognize and categorize human intentions (Chapter 3), and degrees of risk of collision (Chapter 4), in both cases, to adapt the robot's behavior appropriately. For such recognition tasks, we learned that it would not make sense to use ontologies, because reasoning over knowledge would require too much time. In general, collaborative robots would be expected to react quickly to the intention of their collaborators, and the same for any potential safety issue. For this reason, we decided to make the recognition and classification of the different cases directly using data, reducing the amount of potential data to abstract. In the end, we decided to conceptualize knowledge that is useful to *recognize and categorize* events such as collaborations and adaptations, and also knowledge to *decide and choose* between competing plans.

The selection of the ontological language is an essential decision to make, since the language properties will directly affect to the reasoning abilities of the model. In Chapter 5, we used FOL because its expressiveness captured the actual meaning of our conceptualizations, and OWL 2 DL because it allows runtime reasoning in the robot. Of course, the formalization in OWL 2 DL results in a loss of part of the intended meaning, which shall be discussed and even mitigated if possible. For instance, the formal definitions in FOL from that chapter include ternary relationships, which cannot be modeled using OWL 2 DL and would require a workaround (e.g. reification). However, some of the potential limitations could only be solved by using a different language. The logical rules to compare plans from Chapter 7 are not expressible in OWL 2 DL because co-reference of an entity with different roles in an axiom cannot be expressed. This means that if the same entity (e.g. an instantiated plan) appears more than once in an ontological axiom (or rule), there is no way to enforce that both are the same. For this, it was useful to write the proposed rules using FOL syntax and implement them as predicates in Prolog, the language that we used to build the robot's knowledge base. Based on the research presented in this thesis, we think that using different ontological languages is probably a reasonable approach in robotics, since autonomous robots will surely need different types of reasoning. However, this will be an advantage as long as one takes care of the implementation details regarding the integration of the different inferences.

An ontological model captures the semantic structure of the target conceptualization, which can be thought as a graph comprising the semantic relations between the different ontological entities of interest. This thesis postulates that explainable robots shall use ontologies to represent their experiential knowledge as episodic memories, which shall later be visited to retrieve content for an explanation construction. Note that these memories are certainly more powerful than mere data logs, since data is semantically enriched and structured using ontological knowledge. Hence, episodic memories will keep the semantic graphical structure defined in the ontology while adding a temporal dimension. One might wonder how robots might do the retrieval or selection of knowledge depending on the desired explanation. In this thesis, we proposed to leverage the structure of the knowledge, under the assumption that connected knowledge in the graph is semantically relevant to explain that part of the graph. In Chapter 6, we propose a novel algorithm that navigates the knowledge graph's depth to extract the relevant knowledge to narrate collaborative and adaptive robot events. The approach is successfully validated with human users that found the explanations to be useful for understanding robot's experiences. However, even if they are useful and contain all the relevant knowledge, the narratives could be improved by shortening them and generating them using a more human-friendly format. Chapter 7 continues exploring the idea of exploiting the knowledge's structure to build contrastive explanations of competing robot plans. The chapter proposes a novel algorithm that is objectively evaluated against the previous one obtaining positive results. Note that since the principle behind these algorithms is to leverage the knowledge structure, they are general and reusable solutions to build ontology-based explanations even beyond robotic scenarios.

8.2 Challenges and opportunities for future research

8.2.1 Beyond the thesis domain and application scope

It was challenging to model the concept of plan adaptation, but also insightful and inspiring since it plays a relevant role in decision making and explainability for robots beyond collaborative scenarios. There are many concepts related to it that would be worth modeling, but one especially got our attention: the notion of adaptation trigger. The proposed formalization of plan adaptation already provided some intuition about the trigger of the adaptation, however, a deeper investigation in this regard would be needed. Note that the formalization of adaptation trigger would unlock the possibility of constructing new types of ontology-based robot explanations: causal (why did the robot adapt?), or counterfactual (would have the robot adapted if something else had happened?).

8.2.2 Knowledge-based long-term robot memories

Ontology-based episodic memories have proved to be useful for explainable robots, but we think that their true potential remains unexplored. In this thesis, memories were used in relatively short-term tasks, however, episodic memories are one of the types of long-term explicit memory in humans. Hence, it would be interesting to explore the benefits and address the challenges of using them in long-term robot tasks. In those cases, they might be useful for robot introspection for learning how to perform better, or for detecting when it is necessary to improve the learned models. We like to imagine the idea of robots performing tasks during daytime and allowing them to go to sleep to reflect on what can be improved during the night. These ideas seem promising although we understand that building robot memories in the long term would open many issues to attack: what knowledge is relevant to store, when and how robots shall decide to forget, which knowledge to forget, etc. Finally, episodic memories have been used here to construct explanations in this thesis, and we think that explanatory experiences could also be stored in memory. This would help robots to remember what content has already being explained, which can be used to avoid repetitions when explaining similar knowledge.

8.2.3 Knowledge representation formalisms for explainable robots

This work presented in this thesis supports and promotes the use of several formalisms and languages when working with ontologies in robotics. In this regard, we think that there are some formalisms that were not used in our work and would be relevant for robot reasoning tasks in general, and especially useful for explainability purposes. For instance, robots often face uncertainty, especially in human-robot interactive scenarios, and the formalisms used in the thesis fail to model it. For this, one might use non-monotonic logic, which is devised to capture and represent defeasible inferences, i.e., a kind of inference in which reasoners draw tentative conclusions, enabling reasoners to retract their conclusion(s) based on further evidence. Note that this type of logic supports abductive reasoning, a form of logical inference that seeks the simplest and most likely conclusion from a set of observations. We think that such a type of reasoning will be fundamental in the future advances of explainable robots, and this is supported by some works in the literature [Sridharan, 2023].

8.2.4 Ontology-based robot explanations as a social interaction

The focus of our research was on finding effective general methods to retrieve sound knowledge to construct explanations. Hence, the produced explanatory narratives, which were built with the extracted knowledge chunks, were not utterly human-friendly and they would certainly need

to be enhanced. We think that foundational large language models would be of great help with re-writing the textual explanations we built using a more accessible language. Indeed, it would be interesting to use them so that robots could provide explanations interactively, maintaining a dialogue with humans. However, there are some cases in which those models might hallucinate resulting in wrong or false explanations. In this regard, the domain of retrieval augmentation generation (RAG) will surely be source of inspiration for future advancements. This would allow robots to communicate with humans in a natural fashion while providing explanations that are built upon sound ontological knowledge and reasoning.

appendix A

Complete list of publications

This appendix lists all the accepted publications written during the PhD.

A.1 Publications used to write the thesis

- Olivares-Alarcos, A., et al. (2019). *A review and comparison of ontology-based approaches to robot autonomy.* The Knowledge Engineering Review, 34, e29. **Citations in scholar: 135.**

- Olivares-Alarcos, A., Foix, S., and Alenya, G. (2019). *On inferring intentions in shared tasks for industrial collaborative robots.* Electronics, 8(11), 1306. **Citations in scholar: 18.**

- Olivares-Alarcos, A., Foix, S., and Alenyà, G. (2023). *Time-to-contact for robot safety stop in close collaborative tasks.* In book: Human-Robot Collaboration: Unlocking the potential for industrial applications (pp. 87–104). Control, Robotics and Sensors. Institution of Engineering and Technology. **Citations in scholar: 0.**

- Olivares-Alarcos, A., Foix, S., Borgo, S., and Alenyà, G. (2022). *OCRA–An ontology for collaborative robotics and adaptation.* Computers in Industry, 138, 103627. **Citations in scholar: 30. D1 journal.**

- Olivares-Alarcos, A., Andriella, A., Foix, S., and Alenyà, G. (2023). *Robot explanatory narratives of collaborative and adaptive experiences.* In 2023 IEEE International Conference on Robotics and Automation (ICRA) (pp. 11964-11971). **Citations in scholar: 1.**

- Olivares-Alarcos, A., Foix, S., Borràs, J., Canal, G., and Alenyà, G. (2024). *Ontological modeling and reasoning for comparison and contrastive narration of robot plans.* In Proceedings of the 2024 International Conference on Autonomous Agents and Multiagent Systems. **Citations in scholar: 0.**

A.2 Other publications

- Ramos, F., Scrob, C. O., Vázquez, A. S., Fernández, R., and Olivares-Alarcos, A. (2018, October). *Skill-oriented designer of conceptual robotic structures.* In 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 5679-5684). IEEE.

- Kumar, V. R. S., Khamis, A., Fiorini, S., et al. (2019). *Ontologies for industry 4.0.* The Knowledge Engineering Review, 34, e17.

- Gassó Loncan Vallecillo, J., Olivares-Alarcos, A., and Alenyà, G. (2020). *Visual feedback for humans about robots' perception in collaborative environments.* Technical Report IRI-TR-20-03, Institut de Robòtica i Informàtica Industrial, CSIC-UPC.

- Maceira, M., Olivares-Alarcos, A., and Alenya, G. (2020). *Recurrent neural networks for inferring intentions in shared tasks for industrial collaborative robots.* In 2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN) (pp. 665-670). IEEE.

- Olivares-Alarcos, A., Foix, S., and Alenyà, G. (2021). *Knowledge representation for explainability in collaborative robotics and adaptation.* In DAO-XAI'21 - International Workshop on Data meets Applied Ontologies. CEUR-WS.

- Gonçalves, P. J., Olivares-Alarcos, A., Bermejo-Alonso, J., et al. (2021). *IEEE standard for autonomous robotics ontology [standards].* IEEE Robotics & Automation Magazine, 28(3), 171-173.

- Pignaton de Freitas, E., Olszewska, J. I., Carbonera, J. L., et al. (2023). *Ontological concepts for information sharing in cloud robotics.* Journal of Ambient Intelligence and Humanized Computing, 1-12.

appendix B

Complementary material for reviewing ontological models for autonomous robots

This appendix comprises information that complements the work presented in Chapter 2.

B.1 A classification of ontologies for autonomous robots

B.1.1 Ontology scope

Object The Oxford dictionary defines the term `Object` as *a material thing that can be seen and touched*. But often a more fine-grained definition is needed that is not limited to material objects, e.g. holes, or to things in physical space, e.g. behavioral patterns. More generally, mental and social objects depend on material acts (like brain activities and communication acts) but they may be neither material (made of matter) nor physical (located in a region of space). A significant number of foundational ontologies make a distinction between `Endurants` and `Perdurants` [Borgo et al., 2021, Niles and Pease, 2001]. Endurants (aka continuants or objects) are wholly present at any time, but may change over time. Perdurants (aka occurrents or events), on the other hand, are extended in time, and only partially present at any time. This dichotomy is crucial in systems that have to cope with time. Physical objects are often further classified into `Artifact` and `Non-Artifact`, where artifacts are intentionally created, often according to a design, to fulfill a certain function, etc [Borgo et al., 2014]. Objects may further be classified as `Agent` or `Non-Agent`, where agents are capable of generating intentional behavior.

We humans tend to categorize objects because of the variety of `Qualities` and `Properties` that they exhibit, which give us a way to cluster them into similarity classes. There have been long philosophical discussions about what qualities and properties are, and among them which are primary or not, and if they can be exhaustively listed. One important reason to focus on qualities and properties is the understanding of (qualitative) change. One

branch of formal models considers individual qualities (roughly, the way an individual manifests characteristics like weight, size, shape etc.) as basic entities in the ontology. Each individual quality is existentially dependent on a unique endurant (or perdurant) and associated with a quale (plural: qualia). Qualia are used to compare entities, and thus to discuss similarity/dissimilarity (w.r.t. the associated quality) across objects and events [Masolo and Borgo, 2005]. In this view, qualities form a third fundamental category along with endurants and perdurants, and the associated qualia change over time to explain the changes in their corresponding endurants (or perdurants). Qualia are further organized in `Spaces` (e.g. the space of weight, the space of colors etc.) and can be given a quantitative/qualitative value (e.g. numerical) once the space is enriched with a reference system and unit of measure. An alternative approach uses the notion of `Tropes` (also called 'abstract particulars') where qualitative change is expressed through the substitution of tropes [Neuhaus et al., 2004]. Thus, when an object changes, this modeling view assumes that the existing trope ceases to exist and a new one is created. Continuous change (like the increasing of room temperature) is often considered problematic to model in this latter approach.

Environment Map The term `Environment` is defined in the Oxford dictionary as *the surroundings or conditions in which a person, animal, or plant lives or operates* while `Map` is described as *a diagrammatic representation of an area of land or sea showing physical features, cities, roads, etc.* Although some general way to understand the meaning of environment in robotics has been proposed [Borgo et al., 2019a], in this domain the focus is more often oriented to the representation of the environment [Chella et al., 2002]. The format and information content of a map is not only diagrammatic in robotics as it is influenced by how and for what the map is to be used by the robot. Collision maps, for example, encode 3D geometric information of the environment to support generating collision-free motions in 3D space, while navigation maps usually only use a 2D geometrical representation to support finding collision-free navigation paths. The term `Semantic Environment Map` is often used to refer to environment representations that make explicit the semantics of the environment and objects in the environment. Semantic Object Maps encode spatial information about the environment but, in addition, enrich the information content with encyclopedic and common-sense knowledge about objects, and also include knowledge derived from observations [Rusu et al., 2009].

Affordance The term `Affordance` was introduced by Gibson as *what the environment offers the animal, what it provides or furnishes, either for good or ill* [Gibson, 1979]. More recently, the meaning has shifted towards "*(perceived) possibility for action*" [Norman, 2002], i.e., something the object offers that allows the agent to interact with it or, more generally, something that allows objects to participate in actions or processes. However, there is no common agreement in the ontology engineering community on how this concept should be modeled. One way to model affordances is as individual qualities of an object [Ortmann and Kuhn, 2010], or as relational qualities of a pair object-agent [Turvey, 1992]. Another approach is to model them as events, as proposed by Moralez [Moralez, 2023]. A notion of affordance is relevant to talk about possibilities as it enables to answer questions such as "*what can the robot do with an object?*", and "*is it possible for an object to take a particular role when some task is performed?*".

Action and Task There has been a lot of confusion about the meaning of the words `Task` and `Action`, and how these relate to each other. The Oxford dictionary defines an action as *the fact or process of doing something, typically to achieve an aim*. There have been several attempts to define action in different disciplines. A notable one is Donald Davidson's philosophy of action, where he defined an action as something *intentional under some description* [Davidson, 2001]. Krüger and colleagues surveyed the meaning of action in the robotics field [Krüger et al., 2007], and argued that a notion of action in robotics needs to take into account several aspects including perception, actuation, embodiment and learning.

A task can be understood *as a piece of work that has to be done* [OED, 2024]. Hence, tasks denote pending work, independently from how an agent exactly accomplishes this work. In this view, an action would be a way to execute a task. Technically, one can approach this by defining tasks as types (of events) used to classify actions, which then allows one to explicate that a task can be accomplished in different ways, and to talk about individual tasks independently from their possible executions. A notion of task in robotics has been proposed recently by Balakirsky et al. [Balakirsky et al., 2017].

Tasks and actions may further be classified according to their complexity, temporal extension, inter-task (inter-action) relationships, etc. However, such classifications are often not clear, e.g., the distinction between simple vs. complex tasks would be dependent on the adopted granularity or robot's capabilities.

Activity and Behavior The Oxford dictionary defines an `Activity` as *the condition in which things are happening or being done*. The term `Behavior`, on the other hand, is defined as *the way in which one acts or conducts oneself, especially towards others*. Hence, both terms refer to situations in which an agent performs actions, but with different viewpoints. Activities rather having an intrinsic, and behaviors an extrinsic viewpoint, e.g., was it good or bad behavior, how it affected other agents, and so on. Note that in the case of behavior, it can also apply to non-agents as it is common to talk of the behavior of devices or tools, for instance.

In the 80's, Rodney Brooks and his colleagues did fundamental work in the field of *behavior-based robotics* where the term behavior also refers to extrinsic characteristics of task execution. The field of behavior-based robotics is motivated by the observation that complex behavior can be generated by simple control systems, and that intelligence lies in the eye of the observer [Brooks, 1991]. Brooks has also postulated that the world is its own best model, and hence argues that simple *Sense-Act* loops can be used to directly interact with the world without relying much on symbolic representations.

Other authors have focused on the terms `Behavior` and `Function`, for instance claiming that the function of an object denotes its intrinsic aspects (i.e., how it works), and behavior the extrinsic aspects (i.e., what it does). An engineering discussion of this dichotomy is provided by Salustri for the context of computer-based design tools [Salustri, 2000] while an ontological assessment is provided by Mizoguchi et al. [Mizoguchi et al., 2016].

Plan and Method A `Plan` is *a detailed proposal for doing or achieving something* [OED, 2024]. Similarly, the DOLCE+DnS Plan Ontology [Gangemi et al., 2004] defines `Plan` as *a description that defines or uses at least one task and one agentive role or figure, and that has at least one goal as a part*. Hence, plans have explicit goals to be achieved when the plan is executed by appropriate

sequences of actions that comply with the plan. An execution of the plan can succeed, fail, be postponed, aborted, etc.

The generation and assessment of plans is a long-standing sub-area of artificial intelligence. A prominent approach is the Planning Domain Definition Language (PDDL) [McDermott et al., 1998]. PDDL *tasks* denote the initial and goal state, and how the state can be modified by applying actions or operators. General purpose solvers are then used to generate a plan given the domain definition. Several authors have further combined standard planning techniques, such as PDDL, with more expressive representations. A survey about these approaches is provided by Gil [Gil, 2005].

A `Method`, on the other hand, is more abstract than a plan. The Oxford dictionary defines it as *a particular procedure for accomplishing or approaching something, especially a systematic or established one*. In a sense, *methods* are guidelines for agents to choose actions towards achieving a specific goal instead of specifying beforehand an explicit sequence of actions that would cause the goal to be achieved.

Capability and Skill According to the Oxford dictionary, `Capability` is *the power or ability to do something*. Hence, a distinction is made between capabilities that are enabled by physical qualities and those that are enabled by social role(s) within a certain community. The term `Skill`, according to the Oxford dictionary, is more restrictive, namely, *the ability to do something well*. Thus, it only includes what the agent can do because of its physical qualities and, in addition, it implies that the achievement is positively qualified (in terms of manners and results) [Fazel-Zarandi and Fox, 2013]. One widespread use of the term in robotics is *skill learning* where it is used to refer to the ability of the robot to achieve something via a behavior learned through observation, communication, experimentation or simulation. However, both terms are also often used as synonyms of each other, for example by Perzylo and colleagues in their work on the description and orchestration of manufacturing skills [Perzylo et al., 2019a].

Having capabilities represented in a formal model, the robot can reason about whether the necessary capability is present to perform a certain task in a given situational context and, if not, how the task could be accomplished otherwise. This is usually approached by defining capabilities with respect to hardware and software components of the robot [Kunze et al., 2011, Buehler and Pagnucco, 2014]. A navigation capability would be enabled by a mobile base which is controlled by a navigation software component that interfaces with the mobile base. Tiddi et al. have used a notion of capability to provide a more intuitive, capability-based interface to control robots [Tiddi et al., 2017].

Capabilities may not be manifested in arbitrary situations. For example, wheeled robots are not able to navigate along stairs and thus might not be able to reach a target location on another floor. However, if an elevator can be used and the robot is able to operate it, the robot may still be able to reach its navigation goal. Hence, capabilities do not automatically enable the robot to perform a task. Their use depends on suitable conditions of the situational context in which the robot should operate – e.g. who can perform the task, what specific variant of the task can be performed, and where the task can be performed. The degree of how capable a robot is may change over time to the point that a capability cannot be manifested at all, for example, due to attrition of hardware, broken hardware or missing (hardware or software) components.

Hardware components Ontologies may be used to explicate what chains of robot links and joints form what `BodyParts`, and how body parts can contribute to performing, for instance, tasks and capabilities.

One of the most widely used formats to represent hardware components of robotic agents is the *Unified Robot Description Format* (URDF). URDF allows to represent kinematic chains made of links and joints, and also to define the limits of each joint. This information is used, e.g., by inverse kinematics solvers to find a valid joint configuration in which the end-effector of the robot reaches a dedicated goal pose[1]. URDF files include both *actuators* (e.g. servos of the joints, grippers, etc.) which act in the environment and *sensors* (e.g. cameras, sonars, etc.) which used to perceive the environment.

A `Sensor` is a device which detects or measures a physical property and records, indicates, or otherwise responds to it [OED, 2024]. Sensors can be used for different objectives such as measuring robot parameters for control loops, correcting for errors in the robot's models of itself and of the world, and detecting and avoiding failure situations, among others. The Semantic Sensor Network (SSN) is an ontology for describing sensors and their observations, the involved procedures, the studied features of interest, the samples used to do so, and the observed properties, as well as actuators [Compton et al., 2012]. SSN includes a lightweight but self-contained core ontology called SOSA (Sensor, Observation, Sample, and Actuator).

Software components A notion of robot software components in ontologies is crucial when these shall be automatically introspected and integrated into task execution. One of the most widely employed ontologies for modeling software in ontologies is the Ontology of Information Objects (IO) ([Gangemi et al., 2004]) where a distinction is made between an abstract `DataStructure` and the `DigitalResource` that concretely realizes the data structure within some physical storage medium. The broad goals for software ontologies in robotics are to enable the robot's automated software discovery and installation to dynamically compose its control system for a given task, to decide for a given control system whether some capability can be realized by invoking some of the software components, and to support introspection in case some software failure occurred.

One of the most widely used middlewares in robotics nowadays is the Robot Operating System (ROS). ROS organizes robot software components in a communication graph, where each node is a piece of software either listening or publishing messages on named topics, or offering a service that can be called via the node. Messages are defined using an abstract syntax, and concrete realizations of the message type in different target languages such as C++ and Python are generated automatically by ROS.

Interaction and Communication `Interaction` *is a reciprocal action or influence* [OED, 2024] between two or more entities. This comprehensive definition includes those interactions in which there is not an explicit exchange of information. For example, interactions happening at atomic level or *stigmergic* interaction, through the environment in which agents act. Work in robotics tends to concentrate on information exchange. Indeed, in the literature, both the Human-Computer Interaction [Dix, 2009] and the Human-Robot Interaction [Yanco and Drury,

[1] An end-effector is a device located at the end of a kinematic chain, designed to interact with the environment. Its task depends on the application of the robot.

2002, Yanco and Drury, 2004] domains, tend to provide a less general formal definition of `Interaction` which may be closer to `Communication`.

The term `Communication` *is the imparting or exchanging of information by speaking, writing, or using some other medium* [OED, 2024]. Gangemi et al. [Gangemi and Mika, 2003] proposed to formalize this term within the *Description & Situation* Ontology viewpoint distinguishing two cases: an ontology for communication situations and roles, and an ontology for peer-to-peer communication.

B.1.2 Reasoning scope

Recognition and categorization For the purpose of establishing a contact between its environment and its knowledge, a robot must be able to recognize events or situations (static and dynamic) and categorize them as named instances of already known patterns. For instance, let's consider a kitting collaborative robotic task in which a human and a robot aim to fill the compartments of a tray with tokens. The robot must recognize and cateogorize the tray state and the different pieces to manipulate (static), as well as a the human's intentions, the risk of collision with the human, or when a collaboration has finished (dynamic). This thesis presents contributions related to this cognitive capability in Chapters 3, 4, and 5. Note that recognition and categorization are related to perception, since all of them operate on the output generated by perception systems, thus they often are seen as a unique capability. Nevertheless, Langley et al. addressed them separately because they *can individually operate on abstract mental structures* [Langley et al., 2009]. The authors emphasize that in order to support recognition and categorization, a cognitive architecture shall provide a form to represent patterns and situations in memory. This is related to the elements of explainable agency depicted in Fig. 1.2.

Decision making and choice An autonomous robot requires the ability to choose among several alternatives, which usually is considered together with the recognition and categorization problem in a recognize-act cycle. Nonetheless, Langley et al. considered the capability of decision making independently. It is important not to mistake this capability with planning, whose focus is on the achievement of a goal and it will be explained later along this section. For example, a collaborative robot would apply decision making to compare and choose between two competing plans (e.g. one that has a balanced workload or one that is faster but implies a higher robot workload). Meanwhile, planning would be used to find the sequence or sequences of actions for each of those competing plans. Note that cognitive architecture should be able to represent the different choices and their characteristics in a format the robot can understand and easily manipulate to select one option. Indeed, that representation could also be used to improve the decision making through learning, or to construct explanations about the robot's decisions. In this regard, Chapter 7 presents contributions to allow robots to compare alternative plans, decide which one is better based on some criteria, and construct a contrastive explanation comparing them.

Perception and situation assessment The environment where the robot exists, must be sensed, perceived and interpreted. First, the robot senses its surroundings through possibly

muti-modal sensors. Then, the robot is able to perceive the environmental entities (e.g. objects and events), using the gathered information and relying on recognition and categorization, discussed earlier, and on inferential mechanisms, which will be covered shortly. Finally, the situation assessment takes place when the perceived objects and events are interpreted. Following with the example used before, the collaborative robot would look at the tray, the pieces and at the human's movements to sense the environment. The human, its pose and motion, and other information would be recognized and categorized in order to assess the environmental situation so that the robot could, for example, interpret that there is a potential risk of collision with the human and it is needed to adapt. Just as occurred with previous cognitive capabilities, the inherent knowledge of the whole process must be represented in a manner the robot understands. Note that the representation requires memory, a resource which is often limited. Hence, the notion of attention emerges, meaning that the robot not only has to perceive but also it could be asked to decide to focus only on a specific region of the environment or a specific moment in time.

Prediction and monitoring Prediction is a cognitive capability which requires the representation in memory of a model of the environment (e.g. an ontology-based model), the actions that can take place and their effects. Therefore, the robot could predict future events and situations which did not occur yet by means of a proper mechanism which utilizes the representation. Applied to the collaborative robotics' example, it would be possible for the robot to predict the human's intentions, so that the robot could adapt faster to them. Note that prediction enables robots to also monitor processes. When the perceived situation differs from the expected one, it means that either our knowledge is not complete or something did not go as it was supposed to. In the former case, it would be possible to store the facts in memory for posterior learning, in the later case, an alarm or an adaptation could be triggered. In the example, the robot could monitor the risk of collision, or whether a human has stopped collaborating.

Problem solving and planning In novel situations where robots are meant to achieve their goals, it is necessary for them to be able to plan and solve problems. For the purpose of generating a plan, the robot needs a model of the environment utilized to predict the effects of its actions. Furthermore, the cognitive architecture must be able to represent a plan as an (at least partially) ordered set of actions, their expected effects, and the manner in which these effects enable subsequent actions. Sometimes, a robot could have a memory with previous plans which could be re-used with and without further modifications. Note that it is also considered the case of having conditional actions and different branches which depend on the outcome of previous events. Despite often being viewed intimately related, planning is somewhat less general than problem solving. In particular, the former usually refers to cognitive activities within the robot's internal processes, whereas the later can also occur in the world. For instance, when a problem to be solved is complex and the available memory is limited, a robot may search for solutions by executing actions in the environment, rather than constructing a complete internal plan. As an illustration, a collaborative robot could solve a problem by mixing the execution of actions such as: asking for the human's help (external behavior), and the generation of actions' sequences (internal planning).

Reasoning and belief maintenance Reasoning is a cognitive activity which allows a robot to expand its knowledge state, drawing conclusions from other beliefs or assumptions the robot already maintains. Thus, it is required the existence of a representation of beliefs and the relationships among them. A common formalism used to encode such knowledge is FOL. Ontologies are often written in languages based on less expressive formalisms than FOL (e.g. OWL DL) in order to reduce the computational cost of inference. These formalisms, allow the use of different sorts of reasoning such as: deductive or inductive. For the robot of the previous example, it would be possible to infer whether an event is or not a collaboration when a human follows a different plan, or, among alternative plans, which is the best according to some criteria (deductive). Or the opposite, from specific human's behaviors and preferences, inferring the norms to follow during a personalized human-robot interaction (inductive). Note that reasoning is not only relevant to infer new beliefs but also to decide whether to hold existing ones (belief maintenance). Such belief maintenance is especially important for dynamic environments in which situations may change in unexpected ways, with implications for the robot's behavior.

Execution and action Cognition takes place to support and drive activity in the environment. To this end, a cognitive architecture must be able to represent and store motor skills that enable such activity. In the example, a collaborative robotic arm should have skills or policies for manipulating its surroundings and for collaborating or communicating with other agents (e.g. humans). A robot should also be able to execute those skills and actions in the environment, what can happen in a reactive form. Nevertheless, a cognitive architecture should enable a robot to maintain a continuum loop of execution. Hence, the robot would be able to interpret how the execution of actions is affecting the state of the environment and could adapt its behavior. A proper representation of the ongoing actions occurring in the environment, is essential for aspects related to robot action execution: robot adaptation, new skills learning, explaining robot behaviors, etc. Furthermore, the representation should allow to capture the contextual knowledge around executions (e.g. what occurred when the robot was executing a certain action).

Interaction and communication Sometimes, the most effective way for a robot to obtain knowledge is from another agent (e.g. humans, robots, etc.), making communication another important ability that an architecture should support. Going back to the example used before, a collaborative robot could request for further human collaboration to solve a failure, or which are the preferences of the specific human between two alternative plans (for shared decision making). Regardless of the modality or mean of communication, there should be a way to represent the transferred knowledge so that it is accessible to and understandable for the robot. Indeed, this should be bi-directional, meaning the robot must be able to transform stored knowledge into the particular format used for the communication (e.g. knowledge-based construction of textual explanations).

Remembering, reflection, and learning There are some capabilities which cut across those described before, whose use could enhance the performance of autonomous robots while not being strictly necessary for robot autonomy: remembering, reflection and learning.

Remembering is the ability to encode and store the results (facts) of cognitive tasks so that they can be retrieved later. Once again, based on the previous example, a collaborative robot could store the results of an entire day of work (e.g. collaborative and adaptive experiences, decisions about alternative plans, etc.). Reflection stands for the *serious thought or consideration* [OED, 2024] about something which usually is represented and stored in memory and can be retrieved. An example of a reflective process would be the explanation of robot experiences, inferences and decisions in terms of cognitive steps that led to them. Finally, learning, which usually involves generalization beyond specific beliefs and experiences. In the example, the collaborative robot would use the stored memories about successful and failed actions (e.g. adaptations) to generalize and learn from them. The knowledge used to learn might come from distinct sources, the observation of another agent, the result of previous experiences, or through kinesthetic teaching. No matter the source of experience, all of them require the existence of a memory in which the experiences are represented. On this matter, Chapters 6 and 7 investigate the use of ontology-based memories for robot explanation generation.

B.1.3 Application domain scope

Industrial Robots The term of *Industrial Robots* includes all those robots which are automatically controlled, re-programmable, multipurpose manipulator, programmable in three or more axes, which can be either fixed in place or mobile for use in industrial automation applications [ISO 8373:2021, 2021]. Typical applications of industrial robots include welding, painting, assembly, pick and place for printed circuit boards, packaging and labeling, palletizing, product inspection, testing, and material handling. Industrial robots perform with high endurance, speed, and precision in all of those tasks.

Service Robots *Service Robots* are robots in personal use or professional use that perform useful tasks for humans or equipment [ISO 8373:2021, 2021]. Typical applications of service robots include those tasks which are dirty, dull, distant or dangerous. Based on the ISO's definition, service robots can be classified into *service robots for personal* use and *service robots for professional use*. Personal service robots perform tasks such as handling or serving of items, transportation, physical support, providing guidance or information, grooming, cooking and food handling, and cleaning. While professional service robots are meant for inspection, surveillance, handling of items, person transportation, providing guidance or information, cooking and food handling, and cleaning.

B.2 Ontologies to support robot autonomy

B.2.1 Literature search and inclusion criteria

One of the goals of this work is to provide a systematic and fair comparison of projects located at the intersection of the fields autonomous robotics and ontologies. In order to select a potential list of candidates for discussion, we have followed a systematic examination of the state of the art, and, in addition, we have filtered the results by a set of inclusion criteria. In this section, we

discuss the search procedure, and provide a list of criteria that need to be fulfilled by considered approaches.

Literature search

For the purpose of finding literature focused on using ontologies to enhance robot autonomy, we started searching on scientific databases utilizing related keywords. Specifically, we used the literature browser *Web of Science*[2], previously known as *Web of Knowledge*, which is an online subscription-based scientific citation indexing service that provides a comprehensive citation search. It gives access to multiple databases that reference cross-disciplinary research, which allows for in-depth exploration of specialized sub-fields within an academic or scientific discipline.

Typing the keywords *ontology robot autonomy* and *ontology autonomous robotics* yields just 24 and 63 results, respectively. We considered that the number of papers was not enough for our purpose so we went on searching. In the interest of finding a larger list of results, we tried a more general set of keywords: *knowledge representation autonomous robotics*, which returned a list of 306 papers. Going through them, we realized the works were too general, indeed several of them were not even using knowledge representation approaches, therefore, we discarded this list too. The next step was to include the application domain scope in the search (see Section 2.3.3). Hence, the set of keywords was: *knowledge representation industrial robotics* and *knowledge representation service robotics*, with 133 and 148 papers respectively.

The two lists found during the previous step, were combined in a single list of 281 articles in total. In the interest of identifying projects or initiatives that use ontologies to enhance robot autonomy, we have reduced the list of papers following a specific criteria:

- It is proposed to use knowledge representation techniques (ontologies) in robotics applications to enhance robot autonomy;
- the work is part of a project or a big initiative, not just a single article; and
- case studies where the knowledge base is used by a robot exist.

After applying this criterion, the list was reduced to 21 articles, which correspond to five different projects: KnowRob [Tenorth and Beetz, 2009], IEEE-ORA [Schlenoff et al., 2012], ROSETTA [Stenmark and Malec, 2013], CARESSES [Bruno et al., 2019a], and RehabRobo-Onto [Dogmus et al., 2019]. This set of works, was enlarged by other four ones extracted from one of the surveys [Thosar et al., 2018] explained along Section 2.1. In that work, Thosar et al., reviewed a list of nine works, which, as in our work, were chosen following a systematic search and inclusion criteria. We consider that only five of those nine works fit our purpose but one of them is KnowRob, already included before, thus, we only adopt four: ORO [Lemaignan et al., 2010], RoboBrain [Saxena et al., 2015], OUR-K [Lim et al., 2011], and OMRKF [Suh et al., 2007].

It is worth mentioning that the project IEEE-ORA does not actually provide a complete framework which is available to be used, it consists of just an ontology. Indeed, the ontology developed in the framework of that project contains general concepts of the domain, so that it

[2] https://www.webofknowledge.com/

is not really useful in specific application scenarios. However, we understand that it is relevant enough to be considered in our work, since it aims at standardizing the representation of knowledge in the robotics domain. Therefore, we have tried to identify possible extensions of the original work which have been used under the umbrella of available frameworks. Following a similar approach as before (using the Web of Science's browser), we searched for papers that cited the most cited article related to the project [Schlenoff et al., 2012]. In this case, from the 34 initial works which cite it, only two followed the whole inclusion criteria presented in this section: OROSU [Gonçalves and Torres, 2015] and PMK [Diab et al., 2019]. In Section 2.4, where the projects are compared, we consider them as two individual frameworks. Nevertheless, along Section B.2.2, where the approaches are explained, OROSU and PMK are grouped together.

Inclusion criteria

We have already discussed how we have selected the ten frameworks or projects which are considered to be object of the analysis performed in this work. However, among them, we want to focus only on the discussion and the comparison of the most influential approaches. Hence, this section provides a list of inclusion criteria to refine the list of surveyed projects. In the Section B.2.3, we briefly introduce the excluded approaches and provide some justification for our decision. Projects or frameworks are only considered in the scope of this work if they satisfy all of the following criteria:

1. *Ontology scope:* The project uses an ontology that defines one of the terms that we have identified as particularly relevant for autonomous robotics (see Section 2.3.1);

2. *Reasoning scope:* It uses ontologies to support robots manifesting at least one of the cognitive capabilities that we have discussed earlier (see Section 2.3.2);

3. *Transparency:* It is transparent. Meaning that some material (e.g., websites, publications) is openly available that describes the overall goal of the project, what cognitive capabilities are considered, and how and what ontologies are used;

4. *Curation:* It is maintained. Meaning that recent developments or future plans are evident or at least possible; and

5. *Accessibility:* There exists – at least a prototypical – software that is accessible, and that demonstrates how ontologies are used to support a cognitive capability.

B.2.2 Discussion of frameworks/projects

In this section, we give an overview of the six frameworks/projects that have been subject of study in our review. For each of them, their underlying principles and foundations are discussed, as well as what application domain the system was designed for. We also describe how the frameworks evolved over time, and what impact they have had so far. The selection of the presented projects has been done based on the selection criteria presented in Section B.2.1. To the best of our knowledge, we have included in this section all projects that satisfy these criteria.

KnowRob

KnowRob (Knowledge processing for Robots)[3] is an open source[4] knowledge processing system that is designed for autonomous service robots. It was first introduced in 2009 [Tenorth and Beetz, 2009]. Tenorth and Beetz argued that autonomous robot control demands a knowledge representation and reasoning system that addresses several aspects that are commonly not sufficiently considered in analogous systems in artificial intelligence. One of these aspects is that robots need a more fine-grained action representation. This was discussed, in more detail, in another work where Tenorth and Beetz argued that service robots should be able to cope with (often) shallow and symbolic instructions, and to fill in the gaps to generate detailed, grounded, and (often) real-valued information needed for execution [Tenorth and Beetz, 2017].

Recently, a second generation of the KnowRob system was introduced where the focus has shifted towards the integration of simulation and rendering techniques into a hybrid knowledge processing architecture [Beetz et al., 2018, Haidu et al., 2018]. The rationale is to re-use components of the control program in virtual environments with physics and almost photorealistic rendering, and to acquire experiential knowledge from these sources. Experiential knowledge, called *narrative enabled episodic memory* in KnowRob, is used to draw conclusions about what action parameterization is likely to succeed in the real world (e.g., through learning methods) – this principle is inspired by the simulation theory of cognition [Hesslow, 2012].

KnowRob has also been used in several research initiatives including the European projects RoboHow [Beetz et al., 2016], RoboEarth [Waibel et al., 2011], SAPHARI [Beetz et al., 2015a], and SHERPA [Marconi et al., 2012]. RoboEarth, for example, was a pioneer work to consider exchanging knowledge between robots using the World Wide Web, OWL, and Linked Data principles. It was demonstrated, e.g., how such an infrastructure can be used to execute tasks that were not explicitly planned at design time. More recently, KnowRob has been used by the openEASE web knowledge service which is designed for the acquisition, storage, curation, visualization, and analysis of experiential robot knowledge [Beetz et al., 2015c]. KnowRob plays further a central role in the ongoing collaborative research center *Everyday Activity Science & Engineering* (EASE)[5] which has the goal to uncover principles underlying everyday activities by first acquiring experiential knowledge with different modalities, and second building models that generalize over these modalities [Bateman et al., 2018].

The main programming language used in KnowRob is (SWI) Prolog which has its roots in FOL. SWI Prolog comes with a library to manage RDF triples, which is used by KnowRob to represent explicit knowledge in memory such as facts encoded in OWL ontologies. Initially, KnowRob was deriving its concept definitions from the Cyc ontology. However, only rather shallow symbolic representations were used that were tailored to provide useful information for task execution without enforcing consistency in the knowledge base. In recent years, KnowRob has shifted towards the use of the DOLCE+DnS Ultralite (DUL) ontology and a more careful and principled modeling of foundational concepts for autonomous robotics. Another important principle underlying KnowRob is about how data that already exists in the robot control system

[3] http://knowrob.org/
[4] https://github.com/knowrob/knowrob
[5] https://ease-crc.org/

can be made *knowledgeable* – that is how this data can be integrated into symbolic reasoning. KnowRob employs the notion of *virtual knowledge bases* that are computed on demand using control-level data such as data structures used by the perception and planning component of the robot control system. The computation is out carried by, so called, *computable properties* which are computation methods attached to symbolic relation defined in an ontology.

Without a doubt, KnowRob is one of the most influential knowledge representation and reasoning systems for autonomous robots nowadays. This is evident through many research papers and projects that have been using and extending KnowRob since it was initially released. However, there are a couple of limitations worth mentioning here. First, KnowRob has been using only a very shallow symbolic representation following the principles of behavior-based robotics, and in particular the claim that the world itself is its own best model. But having a lot of information only encoded implicitly in data structures of the control program also creates some problems such as the computational cost of abstraction when symbolic inference is performed. Second, despite its long history, no representational standards were proposed by the KnowRob developers. Finally, even though it is one of the most used systems, and openly available, KnowRob has not yet succeeded in creating a large user community, but still has a huge potential to do so in the future.

ROSETTA

ROSETTA[6] stands for *RObot control for Skilled ExecuTion of Tasks in natural interaction with humans; based on Autonomy, cumulative knowledge and learning*. Its origin can be traced to the European projects SIARAS [Haage et al., 2011] and RoSta[7]. During the development of those projects, a set of ontologies of robot skills was implemented with the goal to create an intelligent support system for reconfiguration and adaptation of robot-based manufacturing cells. Those ontologies evolved throughout the scope of other two European projects, ROSETTA and PRACE [Stenmark and Malec, 2013]. The former gave its name to the current ontology. The ROSETTA ontology[8] has further been employed in the research projects SMERobotics [Perzylo et al., 2019b] and SARAFun [Riva and Riva, 2019]. In these projects, the ontology has been used to enhance cognitive abilities of robots that are required to plan and execute assembly tasks. The core ontology has been reorganized after the initial release [Jacobsson et al., 2016], and new case studies on skill reusability in industrial scenarios [Topp et al., 2018] have been developed.

Originally, the ROSETTA ontology did not rely on any upper ontology, however, for more general terms regarding the robotics domain, it currently uses CORA [Schlenoff et al., 2012]. Since CORA relies on SUMO [Niles and Pease, 2001], one can assume that ROSETTA utilizes SUMO as its upper ontology. Even though SUMO is written in a SUO-KIF, a variant of the Knowledge Interchange Format (KIF), a knowledge representation language, ROSETTA is distributed in OWL. The *Knowledge Integration Framework* [Persson et al., 2010] connects all heterogeneous parts of the ROSETTA system: user GUI, simulation, external knowledge sources, task demonstration and the robot. It is the core of the whole system and its goal is to represent, store, adapt, and distribute knowledge across engineering platforms. The data,

[6] http://www.fp7rosetta.org/
[7] Robot standards and reference architectures Project
[8] https://github.com/jacekmalec/Rosetta_ontology

available in the *AutomationML* data exchange format is abstracted using RDF triples. *AutomationML* is an on-going standard initiative that aims at unifying data representation used by engineering tools.

Along this section, we have provided some links to git repositories which are proof of the availability of both the ontology and some parts of the proposed software. However, there is no unique repository containing all different pieces of the complete system as a whole, which reduces the degree of accessibility. Indeed, some parts of the code (e.g. user GUI to program the robot) have not been found at all. On the other hand, the OWL file does not contain the definitions in natural language of the ontological terms, which would help for a better understanding of the formalization. Regarding the scalability of the system, crucial element of any industrial environment, it is not possible to say much due to the small size of the conducted experiments.

IEEE Standard Ontologies for Robotics and Automation

The 1872–2015 IEEE Standard Ontologies for Robotics and Automation [Schlenoff et al., 2012], was developed in the context of the IEEE ORA WG. This standard defines an overall ontology[9] that allows for the representation of, reasoning about, and communication of knowledge in the robotics and automation domain. This ontology includes key terms as well as their definitions, attributes, constraints, and relationships. Sub-parts of this standard include a linguistic framework, generic concepts (an upper ontology), a methodology to add new concepts, and sub-domain ontologies.

The purpose of the standard is to provide an overall ontology and an associated methodology for knowledge representation and reasoning in robotics and automation, together with the representation of concepts in an initial set of application domains. The standard provides a unified way of representing knowledge and provides a common set of terms and definitions, allowing for unambiguous knowledge transfer among any group of human, robots, and other artificial systems.

The proposed ontology is too general to be useful in advanced applications, indeed creating a complete framework was out of the scope of the ORA working group initiative. Nevertheless, the ontology has still been used, for example, Jorge et al., present a scenario where a human ask for a pen and two robots are meant to cooperate in performing the task of collecting and delivering it [Jorge et al., 2015]. Specifically, one robot grasps the pen and poses it on a mobile platform from which the user is supposed to pick the pen up. Further, non-official extensions of the standard have emerged along the last years. In this section, we give a flavor of some of those extensions and how their use enhances robots' autonomy.

OROSU An Ontology for Robotic Orthopedic Surgery (OROSU) [Gonçalves and Torres, 2015] was developed and then applied for hip resurfacing surgery (e.g., for trimming the femoral head). In this scope, the main goal of the research, related to ontologies, was to build a knowledge-based framework for this surgical scenario, along with a formal definition of components and actions to be performed during the surgery. The developed ontology[10] was

[9] https://github.com/srfiorini/IEEE1872-owl
[10] https://github.com/pbsgoncalves/OROSU

partially based on the 1872–2015 – IEEE Standard Ontologies for Robotics and Automation [Schlenoff et al., 2012]. The work was developed under the HIPROB and ECHORD projects, funded by the Portuguese Science Foundation and the EU-FP7, respectively. The framework is among the first to integrate robotic ontologies in the domain of surgical robotics.

The application ontology OROSU, relies on SNOMED CT [Wang et al., 2002], the CORA ontology [Schlenoff et al., 2012] and the KnowRob framework [Tenorth and Beetz, 2013], which were adopted as the upper and reference ontologies. The formal language used to write the ontology was OWL.

It is partially accessible, but it would be desired to have more available material. Indeed, the ontology lacks of natural language definitions, what makes more difficult to understand the specific meaning of the terms. Moreover, the system seems not to have been used by other researchers apart from the developers.

PMK Perception and Manipulation Knowledge (PMK) [Diab et al., 2019] is a knowledge-based reasoning framework that includes some reasoning processes for autonomous robots to enhance Task and Motion Planning (TAMP) capabilities in the manipulation domain. A perception module can be integrated with the framework to capture a rich semantic description of the scene, knowledge about the physical behavior of the objects, and reasoning about the potential manipulation actions. The reasoning scope of PMK is divided into four parts: reasoning for perception, the reasoning for object features, the reasoning for a situation, and reasoning for planning.

PMK follows the preliminary structure of modeling of OUR-K, which divide the knowledge into three gradual layers called, meta-ontology, ontology-schema, and ontology instance [Lim et al., 2011]. PMK enlarged the OUR-K structure by adding some concepts[11] related to the manipulation domain. Moreover, aiming at being shared and reused, PMK ontology relies on other upper and reference/domain ontologies: SUMO [Niles and Pease, 2001] and CORA [Schlenoff et al., 2012].

PMK is meant facilitate the process of manipulation by providing the required components for task and motion planning such as geometric reasoning, dynamic interactions, manipulation and action constraints. The use of PMK could become useful in the domain of robotic manipulation, however, since the system has been recently published, it has not yet been widely extended among other researchers.

PMK was implemented using ontology web language (OWL). Ontology instances can be asserted using information processed from low-level sensory data. Queries over the PMK are based on SWI-Prolog and its Semantic Web library, which serves for loading and accessing ontologies represented in the OWL using Prolog predicates.

ORO

ORO[12] is a project focused on the implementation of a common representation framework for autonomous robots with special emphasizes on human-robot interaction [Lemaignan et al., 2010]. The proposed framework was meant to enhance robot's interaction with complex and

[11] https://github.com/MohammedDiab1/PMK
[12] https://www.openrobots.org/wiki/oro-server/

human-inhabited environments, were robots are expected to exhibit advanced cognitive skills, such as: object recognition, natural language interaction, task planning with possible dynamic re-planning, ability to cooperate with other robots or humans, etc. The authors stated that, these functions, partially independent from each other, need to share common knowledge of the environment where the robot operates.

The ORO's primary component is the *OpenRobots Common Sense Ontology*[13], which precisely provides an upper set of concepts upon which the robot can add and connect new statements of the world [Lemaignan et al., 2010]. This ontology is built upon the OpenCyc upper ontology[14]. The *knowledge core*[15] ontology is a lightweight version of the ORO ontology, which shares the same objective and functionality as its predecessor.

ORO is mainly written in Java, while, *knowledge core*, its lightweight version, is based on Python. The underlying RDF triples storage is done using the open-source Jena framework, which is used together with the Pellet [Sirin et al., 2007] reasoner. ORO provides several wiki pages with detailed explanations of how to use it[16] and how to extend it[17]. In addition, the OWL file contains some natural language definitions, which facilitates the understanding of the ontology.

CARESSES

CARESSES[18] is an international research project whose goal is to design the first robots that can assist older people and adapt to the culture of the individual they are taking care of [Bruno et al., 2019a], [Bruno et al., 2017]. The robots are expected to help the users in many ways including reminding them to take their medication, encouraging them to keep active, helping them keep in touch with family and friends. Each action should be performed with attention to the older person's customs, cultural practices and individual preferences.

CARESSES's principle aim is built upon four fundamental backbones: (a) transcultural robotic nursing, (b) cultural knowledge representation, (c) culturally sensitive planning and execution and (d) culture-aware human-robot interaction. Cultural knowledge, mainly represented using ontologies, enhances the robotic nursing integrating that knowledge into most of the robot processes (e.g. task planning, task execution, human-robot interaction, etc.). Nevertheless, other methodologies such as fuzzy logic and Bayesian networks, are also employed.

The knowledge based proposed within CARESSES consists of three layers: TBox, CBox and PBox. The former is the usual TBox found in any ontology, containing the statements which describe a conceptualization of the domain by defining different sets of individuals described in terms of their characteristics. The second and third layer stand for the usual ABox, which include TBox-compliant statements about individuals belonging to these sets. In this case, in the CBox the statements are related to cultural knowledge while in the PBox the knowledge is about one single person. As it is structured, the knowledge base aims at enhancing the robot adaptation to

[13] https://www.openrobots.org/wiki/oro-ontology
[14] http://www.opencyc.org/
[15] https://github.com/severin-lemaignan/knowledge_core
[16] https://www.openrobots.org/wiki/oro-server-bindings
[17] https://www.openrobots.org/wiki/oro-server-plugins
[18] http://caressesrobot.org/en/

different cultural elderly people but also to the specific preferences of each individual.

Despite the short life of CARESSES, it has become a prominent application example of how ontologies could enhance the autonomy of robots. Nonetheless, the project still presents some drawbacks. For instance, not all of the implemented solutions are publicly available[19]. On the other hand, the ontology's quality is questionable. In the first place, the OWL file lacks of natural language definitions, which hinders the understanding of it. In the second place, some of the entities seem not to be properly defined in a taxonomic view (e.g. event and object are sub-classes of an entity named *topic*, defined as any topic the robot can talk about). Lastly, we would like to discuss the scalability of this approach, which opens some controversial issues. While being a reasonable way of inferring advantageous information of users' preferences, the proposed usage of cultural knowledge, at bigger scale, might result in robots with a strong bias. Depending on the context, users would probably prefer not to feel they are being prejudge (sometimes unfairly) by a robot.

B.2.3 Excluded frameworks/projects

In this section, we give a flavor of some projects which, while having been considered for our analysis, were discarded following our inclusion criteria (see Section B.2.1). Table B.1 shows which of the criteria are met or not by these projects.

Inclusion Criterion	RoboBrain	OMRKF	OUR-K	REHABROBO
1 *Ontology scope*	no	yes	yes	-
2 *Reasoning scope*	no	yes	yes	no
3 *Transparency*	yes	no	no	yes
4 *Curation*	no	no	no	yes
5 *Accessibility*	yes	no	no	no

Table B.1: Inclusion criteria applied to some excluded projects. *yes* indicates that the criterion is met, *no* that it is not met, and - is written when it is unknown.

RoboBrain RoboBrain[20] releases on the Web a huge robot knowledge base where robots can share their experiences and learn [Saxena et al., 2015]. RoboBrain is a large-scale computational system that learns from publicly available Internet resources (e.g., Wikipedia, WordNet, ImageNet, Freebase, OpenCyc), computer simulations, and real-life robot trials. The knowledge is represented in a graph with thousand of nodes and edges. This project presents a relevant effort towards the use of knowledge in robotics and it is continuously maintained. Nonetheless, the developed framework does not use, nor it seems it is planned to start doing it, a concrete formalization of the knowledge based on ontologies, which makes it eligible to be excluded from our analysis.

OMRKF Ontology-based Multi-layered Robot Knowledge Framework (OMRKF) [Suh et al., 2007], aims at enhancing robots intelligence by integrating low-level data with high-level

[19] https://github.com/Suman7495/Robot-Navigation-for-Vision-Based-HAR
[20] http://robobrain.me/index.html

knowledge into the same framework. OMRKF has four levels of knowledge, each of them split into three levels: (a) model (object feature, object and space); (b) context (spatial, temporal and high-level); (c) perception (numerical descriptor, visual feature and visual concept); and (d) activity (behavior, task and service). In this context, knowledge representation helps robots to execute sequenced behaviors by just specifying the high level service and also how the robot can recognize objects even when the knowledge is not complete. This system means a good effort towards cognitive autonomous robots. However, it has not been possible to find any available material nor enough documentation, hence, the project is excluded.

OUR-K Ontology-based Unified Robot Knowledge (OUR-K) is a framework that integrates low-level data with high-level knowledge for robot intelligence in service robotics scenarios [Lim et al., 2011]. It seems to be an extension of OMRKF, because they share some similarities and authors. The framework consist of three parts: (a) knowledge description, (b) knowledge association, and (c) application. The former takes care of the representation of knowledge (low-level data and high-level knowledge) by using five classes of entities: features, objects, spaces, contexts and actions. Knowledge association specifies the relationships between different descriptions allowing several inference methods (logics, bayesian inference, heuristics). Finally, the descriptions and their relationships are used in several applications: navigation, action selection, object recognition, context awareness, planning and object manipulation. Lim [Lim, 2019], presents how OUR-K is used in some case studies where the knowledge the robot holds is incomplete. This framework seems to be beneficial to our domain and it could have been included in our analysis if it were not for the absence of available resources. None the ontology nor the framework have been found in public repositories.

REHABROBO REHABROBO-QUERY [Dogmus et al., 2019] is a web based software which allows robot designers to add, modify and consult information about their rehabilitation robots. This information is stored using a the formal ontology REHABROBO-ONTO [Dogmus et al., 2015]. The whole system is available on the cloud, utilizing Amazon EC2. The whole framework turns into a useful tool when your intention is to store/consult descriptions of rehabilitation robots (e.g. robot parts, capabilities, etc.). Nevertheless, it remains unclear how this system could be utilized to equip robots with autonomy, indeed, the system has not been used to solve any of the reasoning problems proposed in Section 2.3.2. Moreover, there is no simple way of accessing to the OWL file of the ontology and the Amazon server they use is not fully free. It was decided then, not to include this work in our analysis.

appendix C

Pilot study questionnaire

In Chapter 6, the quality of information measurement discussed by Lee et al. [Lee et al., 2002] was used to evaluate the narratives. They presented a model for Information Quality, a questionnaire to measure it, and analysis techniques to interpret the measures. We used one of the quadrants of their model and its relative questionnaire: *usefulness*. It aims to assess whether or not the information is relevant to the user's task, in our case, the 'new operator training task'. In particular, *usefulness* was measured through five dimensions: *appropriate amount*, *relevancy*, *understandability*, *interpretability*, and *objectivity*. For each dimension, a set of questions had to be evaluated using an 11-point Likert scale ranging from completely disagree (0) to completely agree (10). Note that the points of items labeled with '(R)' shall be reversed, and that we also added some qualitative measures to the questionnaire in the form of four open questions.

C.1 Quantitative measures

C.1.1 Appropriate Amount

(4 items, Cronbach's Alpha = .76)

- This information is of sufficient volume for our needs.
- The amount of information does not match our needs. (R)
- The amount of information is not sufficient for our needs. (R)
- The amount of information is neither too much nor too little.

C.1.2 Relevancy

(4 items, Cronbach's Alpha = .94)

- This information is useful to our work.
- This information is relevant to our work.
- This information is appropriate for our work.
- This information is applicable to our work.

C.1.3 Understandability

(4 items, Cronbach's Alpha = .90)

- This information is easy to understand.
- The meaning of this information is difficult to understand. (R)
- This information is easy to comprehend.
- The meaning of this information is easy to understand.

C.1.4 Interpretability

(5 items, Cronbach's Alpha = .77)

- It is easy to interpret what this information means.
- This information is difficult to interpret. (R)
- It is difficult to interpret the coded information. (R)
- This information is easily interpretable.
- The measurement units for this information are clear.

C.1.5 Objectivity

(4 items, Cronbach's Alpha = .72)

- This information was objectively collected.
- This information is based on facts.
- This information is objective.
- This information presents an impartial view.

C.2 Qualitative measures

1. Imagine that you are asked to collaborate with the real robot. This includes to understand the robot's plan adaptations and the collaborative goal and plan. Do you believe that a video without explanations can be enough to prepare and train you to collaborate with the robot? **(Y/N)** Please, explain your answer.

2. Continue imagining that you are asked to collaborate with the real robot. Do you think that the textual explanations have helped you to be prepared for the different situations that you can face during the collaboration? **(Y/N)** Please, explain your answer.

3. Now imagine that when an explanation has something in common with a previous explanation, you receive a summarized explanation. Would you prefer a summarized (shorter) explanation or a complete (longer) one that is more informative but repeats some information? **(Summarized/Complete)** Please, explain your answer.

4. Think for a moment that you can select the content of the explanations about the plan adaptations and the collaborations. Which content would you choose so that you could better learn to collaborate with the robot?

appendix D

Additional explanatory narratives

This appendix includes the automatically generated narrative of each of the robot experiences used in Chapter 6 for the evaluation with users. The narratives were constructed with specificity level 3, which also includes the result of levels 1 (red) and 2 (blue).

D.1 Event 28

An example of collaboration, the shared goal was to have a full board with tokens with odd numbers. The narrative is:

'Event_28' is a type of 'Collaboration' from 300.0 to 340.0 and is classified by 'Collaboration Class' and has quality 'Current Risk Of Collision' and has participant 'Human and Robot' and executes plan 'Place Tokens With Odd Numbers' and has location 'pm Lab'. 'Current Risk Of Collision' is a type of 'Collaboration Risk' and has data value 'Low Risk' from 300.0 to 312.0 and has data value 'Low Risk' from 314.0 to 340.0 and has data value 'Medium Risk' from 312.0 to 314.0. 'Human' is a type of 'Physical Agent' from 1.0 to 1000.0 and has goal 'Full Board With Tokens With Odd Numbers' and has plan 'Place Tokens With Odd Numbers'. 'Robot' is a type of 'Physical Agent' from 1.0 to 1000.0 and has plan 'Place Tokens With Odd Numbers' and has goal 'Full Board With Tokens With Odd Numbers'. 'Place Tokens With Odd Numbers' is a type of 'Plan' from 1.0 to 1000.0 and has component 'Full Board With Tokens With Odd Numbers'. 'Collaboration Class' is a type of 'Indirectly Physical Collaboration'. 'pm Lab' is a type of 'Collaboration Place' from 1.0 to 1000.0.

D.2 Event 30

An example of collaboration, the shared goal was to have a full board with tokens in ascending order. The narrative is:

'Event_30' is a type of 'Collaboration' from 400.0 to 440.0 and is classified by 'Collaboration Class' and has quality 'Current Risk Of Collision' and has participant 'Human and Robot' and executes plan 'Place Tokens In Ascending Order' and has location 'pm Lab'. 'Current Risk Of Collision' is a type of 'Collaboration Risk' and has data value 'Low Risk' from 400.0 to 432.0 and has data value 'Low Risk' from 436.0 to 440.0 and has data value 'Medium Risk' from 432.0 to 436.0. 'Human' is a type of 'Physical Agent' from 1.0 to 1000.0 and has goal 'Full Board With Tokens In Ascending Order' and has plan 'Place Tokens In Ascending Order'. 'Robot' is a type of 'Physical Agent' from 1.0 to 1000.0 and has plan 'Place Tokens In Ascending Order' and has goal 'Full Board With Tokens In Ascending Order'. 'Place Tokens In Ascending Order' is a type of 'Plan' from 1.0 to 1000.0 and has component 'Full Board With Tokens In Ascending Order'. 'Collaboration Class' is a type of 'Indirectly Physical Collaboration'. 'pm Lab' is a type of 'Collaboration Place' from 1.0 to 1000.0.

D.3 Event 33

An example of collaboration, the shared goal was to have a full board with tokens by color in columns. The narrative is:

'Event_33' is a type of 'Collaboration' from 500.0 to 542.0 and is classified by 'Collaboration Class' and has quality 'Current Risk Of Collision' and has participant 'Human and Robot' and executes plan 'Place Tokens In Columns By Color' and has location 'pm Lab'. 'Current Risk Of Collision' is a type of 'Collaboration Risk' and has data value 'Low Risk' from 500.0 to 508.0 and has data value 'Low Risk' from 510.0 to 542.0 and has data value 'Medium Risk' from 508.0 to 510.0. 'Human' is a type of 'Physical Agent' from 1.0 to 1000.0 and has goal 'Full Board With Tokens In Columns By Color' and has plan 'Place Tokens In Columns By Color'. 'Robot' is a type of 'Physical Agent' from 1.0 to 1000.0 and has plan 'Place Tokens In Columns By Color' and has goal 'Full Board With Tokens In Columns By Color'. 'Place Tokens In Columns By Color' is a type of 'Plan' from 1.0 to 1000.0 and has component 'Full Board With Tokens In Columns By Color'. 'Collaboration Class' is a type of 'Indirectly Physical Collaboration'. 'pm Lab' is a type of 'Collaboration Place' from 1.0 to 1000.0.

D.4 Event 9

An example of non-collaboration, the human stopped participating in the event to start taking notes. The narrative is:

'Event_9' (not) is a type of 'Collaboration' and is a type of 'Event' from 1.0 to 41.0 and executes plan 'Place Tokens In Columns By Color' and has participant 'Robot' and (not) has participant 'Human'. 'Place Tokens In Columns By Color' is a type of 'Plan' from 1.0 to 1000.0 and has component 'Full Board With Tokens In Columns By Color' and is plan of 'Human and Robot'. 'Robot' is a type of 'Physical Agent' from 1.0 to 1000.0 and has goal 'Full Board With Tokens In Columns By Color'. 'Human' is a type of 'Physical Agent' from 1.0 to 1000.0 and has goal 'Full Board With Tokens In Columns By Color'.

D.5 Event 15

An example of non-collaboration, the human stopped participating in the event to leave the workspace. The narrative is:

> 'Event_15' (not) is a type of 'Collaboration' and is a type of 'Event' from 100.0 to 142.0 and executes plan 'Place Tokens In Columns By Color' and has participant 'Robot' and (not) has participant 'Human'. 'Place Tokens In Columns By Color' is a type of 'Plan' from 1.0 to 1000.0 and has component 'Full Board With Tokens In Columns By Color' and is plan of 'Robot and Human'. 'Robot' is a type of 'Physical Agent' from 1.0 to 1000.0 and has goal 'Full Board With Tokens In Columns By Color'. 'Human' is a type of 'Physical Agent' from 1.0 to 1000.0 and has goal 'Full Board With Tokens In Columns By Color'.

D.6 Event 27

An example of non-collaboration, the human stopped executing the shared plan (filling in ascending order) to start executing a different plan (filling by colors in columns). The narrative is:

> 'Event_27' (not) is a type of 'Collaboration' and is a type of 'Event' from 200.0 to 240.0 and has participant 'Human and Robot' and executes plan 'Place Tokens In Ascending Order and Place Tokens In Columns By Color'. 'Human' is a type of 'Physical Agent' from 1.0 to 1000.0 and has goal 'Full Board With Tokens In Columns By Color' and has plan 'Place Tokens In Columns By Color' and (not) has plan 'Place Tokens In Ascending Order' and (not) has goal 'Full Board With Tokens In Ascending Order'. 'Robot' is a type of 'Physical Agent' from 1.0 to 1000.0 and has plan 'Place Tokens In Ascending Order' and has goal 'Full Board With Tokens In Ascending Order'. 'Place Tokens In Ascending Order' is a type of 'Plan' from 1.0 to 1000.0 and has component 'Full Board With Tokens In Ascending Order'. 'Place Tokens In Columns By Color' is a type of 'Plan' from 1.0 to 1000.0 and has component 'Full Board With Tokens In Columns By Color'.

D.7 Event 39

An example of plan adaptation, the robot issued a safety stop due to a high risk of collision. The narrative is:

'Event_39' is a type of 'Plan Adaptation' from 600.0 to 613.0 and has part 'Execution Of Place Token On Compartment19' from 600.0 to 607.0 and has part 'Execution Of Stop Until Human Command' from 607.5 to 613.0 and has participant 'Robot'. 'Execution Of Stop Until Human Command' is a type of 'Event' from 607.5 to 613.0 and is postcondition of 'High Collision Risk' from 607.5 to 613.0 and has participant 'Robot' from 607.5 to 613.0 and executes plan 'Stop Until Human Command' from 607.5 to 613.0 and (not) executes plan 'Place Token On Compartment19' from 607.5 to 613.0. 'Robot' is a type of 'Physical Agent' from 1.0 to 1000.0 and has goal 'Full Board With Tokens In Columns By Color' from 600.0 to 649.0 and has plan 'Place Token On Compartment19' from 600.0 to 607.0 and has plan 'Place Tokens In Columns By Color' from 600.0 to 649.0 and has plan 'Stop Until Human Command' from 607.5 to 613.0. 'Execution Of Place Token On Compartment19' is a type of 'Event' from 600.0 to 607.0 and is precondition of 'High Collision Risk' from 607.0 to 613.0 and executes plan 'Place Token On Compartment19' from 600.0 to 607.0 and has participant 'Robot' from 600.0 to 607.0. 'Event_38' has participant 'Robot' from 600.0 to 649.0.

D.8 Event 43

An example of plan adaptation, the robot discarded the token to the trash because the number on the token was too small according to the current tokens on the board. The human has placed a token on the board that triggered the adaptation. The narrative is:

'Event_43' is a type of 'Plan Adaptation' from 650.0 to 665.0 and has part 'Execution Of Place Token On Compartment20' from 650.0 to 658.0 and has part 'Execution Of Place Token On Trash' from 658.5 to 665.0 and has participant 'Robot'. 'Execution Of Place Token On Trash' is a type of 'Event' from 658.5 to 665.0 and executes plan 'Place Token On Trash' from 658.5 to 665.0 and has participant 'Robot' from 658.5 to 665.0 and is postcondition of 'Token Number Is Too Small' from 658.5 to 665.0 and (not) executes plan 'Place Token On Compartment20' from 658.5 to 665.0. 'Robot' is a type of 'Physical Agent' from 1.0 to 1000.0 and has goal 'Full Board With Tokens In Descending Order' from 650.0 to 699.0 and has plan 'Place Token On Compartment20' from 650.0 to 658.0 and has plan 'Place Token On Trash' from 658.5 to 665.0 and has plan 'Place Tokens In Descending Order' from 650.0 to 699.0. 'Execution Of Place Token On Compartment20' is a type of 'Event' from 650.0 to 658.0 and executes plan 'Place Token On Compartment20' from 650.0 to 658.0 and has participant 'Robot' from 650.0 to 658.0 and is precondition of 'Token Number Is Too Small' from 658.0 to 665.0. 'Event_42' has participant 'Robot' from 650.0 to 699.0.

D.9 Event 49

An example of plan adaptation, the robot changed its target compartment because the human filled it. The narrative is:

'Event_49' is a type of 'Plan Adaptation' from 700.0 to 715.0 and has part 'Execution Of Place Token On Compartment15' from 707.5 to 715.0 and has part 'Execution Of Place Token On Compartment19' from 700.0 to 707.0 and has participant 'Robot'. 'Execution Of Place Token On Compartment15' is a type of 'Event' from 707.5 to 715.0 and executes plan 'Place Token On Compartment15' from 707.5 to 715.0 and has participant 'Robot' from 707.5 to 715.0 and is postcondition of 'Target Compartment Is Full' from 707.5 to 715.0 and (not) executes plan 'Place Token On Compartment19' from 707.5 to 715.0. 'Robot' is a type of 'Physical Agent' from 1.0 to 1000.0 and has goal 'Full Board With Tokens In Quadrants By Color' from 700.0 to 749.0 and has plan 'Place Token On Compartment15' from 707.5 to 715.0 and has plan 'Place Token On Compartment19' from 700.0 to 707.0 and has plan 'Place Tokens In Quadrants By Color' from 700.0 to 749.0. 'Execution Of Place Token On Compartment19' is a type of 'Event' from 700.0 to 707.0 and executes plan 'Place Token On Compartment19' from 700.0 to 707.0 and has participant 'Robot' from 700.0 to 707.0 and is precondition of 'Target Compartment Is Full' from 707.0 to 715.0. 'Event_48' has participant 'Robot' from 700.0 to 749.0.

D.10 Event 51

An example of plan adaptation, the robot placed the token on the auxiliary pile for later use because the target compartment is busy with an incorrect token. Note that the human is expected to pick and place the incorrectly placed token (freeing the compartment) because the robot cannot reach the pose where it should go. The narrative is:

'Event_51' is a type of 'Plan Adaptation' from 750.0 to 763.5 and has part 'Execution Of Place Token On Auxiliar Pile' from 757.5 to 763.5 and has part 'Execution Of Place Token On Compartment19' from 750.0 to 757.0 and has participant 'Robot'. 'Execution Of Place Token On Auxiliar Pile' is a type of 'Event' from 757.5 to 763.5 and executes plan 'Place Token On Auxiliar Pile' from 757.5 to 763.5 and has participant 'Robot' from 757.5 to 763.5 and is postcondition of 'Target Color Is Full With Wrong Token' from 757.5 to 763.5 and (not) executes plan 'Place Token On Compartment19' from 757.5 to 763.5. 'Robot' is a type of 'Physical Agent' from 1.0 to 1000.0 and has goal 'Full Board With Tokens In Quadrants By Color' from 750.0 to 799.0 and has plan 'Place Token On Auxiliar Pile' from 757.5 to 763.5 and has plan 'Place Token On Compartment19' from 750.0 to 757.0 and has plan 'Place Tokens In Quadrants By Color' from 750.0 to 799.0. 'Execution Of Place Token On Compartment19' is a type of 'Event' from 750.0 to 757.0 and executes plan 'Place Token On Compartment19' from 750.0 to 757.0 and has participant 'Robot' from 750.0 to 757.0 and is precondition of 'Target Color Is Full With Wrong Token' from 757.0 to 763.5. 'Event_50' has participant 'Robot' from 750.0 to 799.0.

D.11 Event 59

An example of plan adaptation, the robot discarded the held token to the trash because the number on the token was too large according to the current tokens on the board. The narrative is:

'Event_59' is a type of 'Plan Adaptation' from 800.0 to 820.0 and has part 'Execution Of Place Token On Compartment20' from 800.0 to 812.0 and has part 'Execution Of Place Token On Trash' from 812.5 to 820.0 and has participant 'Robot'. 'Execution Of Place Token On Trash' is a type of 'Event' from 812.5 to 820.0 and executes plan 'Place Token On Trash' from 812.5 to 820.0 and has participant 'Robot' from 812.5 to 820.0 and is postcondition of 'Token Number Is Too High' from 812.5 to 820.0 and (not) executes plan 'Place Token On Compartment20' from 812.5 to 820.0. 'Robot' is a type of 'Physical Agent' from 1.0 to 1000.0 and has goal 'Full Board With Tokens In Descending Order' from 800.0 to 849.0 and has plan 'Place Token On Compartment20' from 800.0 to 812.0 and has plan 'Place Token On Trash' from 812.5 to 820.0 and has plan 'Place Tokens In Descending Order' from 800.0 to 849.0. 'Execution Of Place Token On Compartment20' is a type of 'Event' from 800.0 to 812.0 and executes plan 'Place Token On Compartment20' from 800.0 to 812.0 and has participant 'Robot' from 800.0 to 812.0 and is precondition of 'Token Number Is Too High' from 812.0 to 820.0. 'Event_58' has participant 'Robot' from 800.0 to 849.0.

D.12 Event 63

An example of plan adaptation, the robot changed its target compartment because the human filled it. The narrative is:

'Event_63' is a type of 'Plan Adaptation' from 850.0 to 868.0 and has part 'Execution Of Place Token On Compartment18' from 850.0 to 858.0 and has part 'Execution Of Place Token On Compartment19' from 858.5 to 868.0 and has participant 'Robot'. 'Execution Of Place Token On Compartment19' is a type of 'Event' from 858.5 to 868.0 and executes plan 'Place Token On Compartment19' from 858.5 to 868.0 and has participant 'Robot' from 858.5 to 868.0 and is postcondition of 'Target Compartment Is Full' from 858.5 to 868.0 and (not) executes plan 'Place Token On Compartment18' from 858.5 to 868.0. 'Robot' is a type of 'Physical Agent' from 1.0 to 1000.0 and has goal 'Full Board With Tokens In Descending Order' from 850.0 to 899.0 and has plan 'Place Token On Compartment18' from 850.0 to 858.0 and has plan 'Place Token On Compartment19' from 858.5 to 868.0 and has plan 'Place Tokens In Descending Order' from 850.0 to 899.0. 'Execution Of Place Token On Compartment18' is a type of 'Event' from 850.0 to 858.0 and executes plan 'Place Token On Compartment18' from 850.0 to 858.0 and has participant 'Robot' from 850.0 to 858.0 and is precondition of 'Target Compartment Is Full' from 858.0 to 868.0. 'Event_62' has participant 'Robot' from 850.0 to 899.0.

Bibliography

[2021/0106(COD), 2024] 2021/0106(COD) (2024). Proposal for a Regulation of the European Parliament and of the Council laying down harmonised rules on artificial intelligence (Artificial Intelligence Act) and amending certain Union legislative acts - Analysis of the final compromise text with a view to agreement. Technical report, European Commission. (Cited on p. 4)

[Ajoudani et al., 2018] Ajoudani, A., Zanchettin, A. M., Ivaldi, S., Albu-Schäffer, A., Kosuge, K., and Khatib, O. (2018). Progress and prospects of the human–robot collaboration. *Autonomous Robots*, 42(5):957–975. (Cited on p. 89, 90)

[Alenyà et al., 2009] Alenyà, G., Nègre, A., and Crowley, J. L. (2009). A comparison of three methods for measure of time to contact. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4565–4570. (Cited on p. 67)

[Androutsopoulos et al., 2013] Androutsopoulos, I., Lampouras, G., and Galanis, D. (2013). Generating natural language descriptions from owl ontologies: the naturalowl system. *Journal of Artificial Intelligence Research*, 48:671–715. (Cited on p. 112)

[Anjomshoae et al., 2019] Anjomshoae, S., Najjar, A., Calvaresi, D., and Främling, K. (2019). Explainable agents and robots: Results from a systematic literature review. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, page 1078–1088. International Foundation for Autonomous Agents and Multiagent Systems. (Cited on p. 5, 110)

[Arp et al., 2015] Arp, R., Smith, B., and Spear, A. D. (2015). *Building Ontologies with Basic Formal Ontology*. MIT Press. (Cited on p. 19)

[Bagnall et al., 2017] Bagnall, A., Lines, J., Bostrom, A., Large, J., and Keogh, E. (2017). The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data Mining and Knowledge Discovery*, 31(3):606–660. (Cited on p. 48)

[Balakirsky, 2015] Balakirsky, S. (2015). Ontology based action planning and verification for agile manufacturing. *Robotics and Computer-Integrated Manufacturing*, 33:21 – 28. (Cited on p. 86, 112)

[Balakirsky et al., 2017] Balakirsky, S., Schlenoff, C., Rama Fiorini, S., Redfield, S., Barreto, M., Nakawala, H., Carbonera, J. L., Soldatova, L., Bermejo-Alonso, J., Maikore, F., Goncalves, P.

J. S., De Momi, E., Sampath Kumar, V. R., and Haidegger, T. (2017). Towards a Robot Task Ontology Standard. In *Proceedings of the ASME 2017 12th International Manufacturing Science and Engineering Conference*, volume Volume 3: Manufacturing Equipment and Systems, page V003T04A049. (Cited on p. 163)

[Barandiaran and Moreno, 2008] Barandiaran, X. and Moreno, A. (2008). Adaptivity: From metabolism to behavior. *Adaptive Behavior*, 16(5):325–344. (Cited on p. 93)

[Barandiaran et al., 2009] Barandiaran, X. E., Paolo, E. D., and Rohde, M. (2009). Defining agency: Individuality, normativity, asymmetry, and spatio-temporality in action. *Adaptive Behavior*, 17(5):367–386. (Cited on p. 93)

[Barreiro et al., 2009] Barreiro, J., Jones, G., and Schaffer, S. (2009). Peer-to-peer planning for space mission control. In *2009 IEEE Aerospace conference*, pages 1–9. (Cited on p. 147)

[Bartels et al., 2019] Bartels, G., Beßler, D., and Beetz, M. (2019). Episodic memories for safety-aware robots. *KI - Künstliche Intelligenz*, 33(2):123–130. (Cited on p. 112)

[Bastianelli et al., 2014] Bastianelli, E., Castellucci, G., Croce, D., Iocchi, L., Basili, R., and Nardi, D. (2014). HuRIC: a human robot interaction corpus. In Calzolari, N., Choukri, K., Declerck, T., Loftsson, H., Maegaard, B., Mariani, J., Moreno, A., Odijk, J., and Piperidis, S., editors, *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC)*, pages 4519–4526. European Language Resources Association (ELRA). (Cited on p. 41)

[Bateman et al., 2018] Bateman, J., Beetz, M., Beßler, D., Bozcuoğlu, A. K., and Pomarlan, M. (2018). Heterogeneous ontologies and hybrid reasoning for service robotics: The ease framework. In Ollero, A., Sanfeliu, A., Montano, L., Lau, N., and Cardeira, C., editors, *ROBOT 2017: Third Iberian Robotics Conference*, pages 417–428, Cham. Springer International Publishing. (Cited on p. 172)

[Bauer et al., 2008] Bauer, A., Wollherr, D., and Buss, M. (2008). Human–robot collaboration: a survey. *International Journal of Humanoid Robotics*, 5(01):47–66. (Cited on p. 89, 90)

[Bauer et al., 2016] Bauer, W., Bender, M., Braun, M., Rally, P., and Scholtz, O. (2016). Lightweight robots in manual assembly—best to start simply. Technical report, Frauenhofer-Institut für Arbeitswirtschaft und Organisation IAO, Stuttgart. (Cited on p. 40, 95)

[Beer et al., 2014] Beer, J. M., Fisk, A. D., and Rogers, W. A. (2014). Toward a framework for levels of robot autonomy in human-robot interaction. *Journal of human-robot interaction*, 3(2):74–99. (Cited on p. 20)

[Beetz et al., 2015a] Beetz, M., Bartels, G., Albu-Schäffer, A., Bálint-Benczédi, F., Belder, R., Beßler, D., Haddadin, S., Maldonado, A., Mansfeld, N., Wiedemeyer, T., Weitschat, R., and Worch, J.-H. (2015a). Robotic agents capable of natural and safe physical interaction with human co-workers. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6528–6535. (Cited on p. 172)

[Beetz et al., 2018] Beetz, M., Beßler, D., Haidu, A., Pomarlan, M., Bozcuoğlu, A. K., and Bartels, G. (2018). Know rob 2.0 — a 2nd generation knowledge processing framework for cognition-enabled robotic agents. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 512–519. (Cited on p. 29, 34, 86, 87, 98, 110, 112, 113, 124, 137, 140, 172)

[Beetz et al., 2020] Beetz, M., Beßler, D., Koralewski, S., Pomarlan, M., Vyas, A., Hawkin, A., Dhanabalachandran, K., and Jongebloed, S. (2020). Neem handbook. [online], Institute for Artificial Intelligence (IAI), University of Bremen. https://ease-crc.github.io/soma/owl/current/NEEM-Handbook.pdf. (Cited on p. 114)

[Beetz et al., 2016] Beetz, M., Beßler, D., Winkler, J., Worch, J.-H., Bálint-Benczédi, F., Bartels, G., Billard, A., Bozcuoğlu, A. K., Fang, Z., Figueroa, N., Haidu, A., Langer, H., Maldonado, A., Ureche, A. L. P., Tenorth, M., and Wiedemeyer, T. (2016). Open robotics research using web-based knowledge services. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5380–5387. (Cited on p. 172)

[Beetz et al., 2015b] Beetz, M., Bálint-Benczédi, F., Blodow, N., Nyga, D., Wiedemeyer, T., and Márton, Z.-C. (2015b). Robosherlock: Unstructured information processing for robot perception. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1549–1556. (Cited on p. 29, 30)

[Beetz et al., 2010] Beetz, M., Mösenlechner, L., and Tenorth, M. (2010). Cram — a cognitive robot abstract machine for everyday manipulation in human environments. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1012–1017. (Cited on p. 29, 33)

[Beetz et al., 2015c] Beetz, M., Tenorth, M., and Winkler, J. (2015c). Open-ease – a knowledge processing service for robots and robotics/ai researchers. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1983–1990. (Cited on p. 29, 34, 172)

[Bermejo-Alonso, 2018] Bermejo-Alonso, J. (2018). Reviewing task and planning ontologies: An ontology engineering process. In *International Conference on Knowledge Engineering and Ontology Development (KEOD)*, pages 181–188. (Cited on p. 129)

[Berndt and Clifford, 1994] Berndt, D. J. and Clifford, J. (1994). Using dynamic time warping to find patterns in time series. In *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*, page 359–370. AAAI Press. (Cited on p. 48)

[Beßler et al., 2018] Beßler, D., Pomarlan, M., and Beetz, M. (2018). Owl-enabled assembly planning for robotic agents. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, page 1684–1692. International Foundation for Autonomous Agents and Multiagent Systems. (Cited on p. 29, 31, 32)

[Beßler et al., 2020a] Beßler, D., Porzel, R., Mihai, P., Beetz, M., Malaka, R., and Bateman, J. (2020a). A formal model of affordances for flexible robotic task execution. In *Proceedings of the 24th European Conference on Artificial Intelligence (ECAI)*. (Cited on p. 25)

[Beßler et al., 2023] Beßler, D., Porzel, R., Pomarlan, M., and Beetz, M. (2023). Foundational models for manipulation activity parsing. In Jung, T., tom Dieck, M. C., and Correia Loureiro, S. M., editors, *Extended Reality and Metaverse*, pages 115–121. Springer International Publishing. (Cited on p. 28, 29)

[Beßler et al., 2020b] Beßler, D., Porzel, R., Pomarlan, M., Beetz, M., Malaka, R., and Bateman, J. (2020b). A formal model of affordances for flexible robotic task execution. In *Proceedings of the 24th European Conference on Artificial Intelligence (ECAI 2020)*, pages 2425–2432. (Cited on p. 112)

[Borgo, 2019] Borgo, S. (2019). An ontological view of components and interactions in behaviorally adaptive systems. *J. Integr. Des. Process. Sci.*, 23(1):17–35. (Cited on p. 90)

[Borgo et al., 2019a] Borgo, S., Cesta, A., Orlandini, A., and Umbrico, A. (2019a). Knowledge-based adaptive agents for manufacturing domains. *Engineering with Computers*, 35(3):755–779. (Cited on p. 85, 86, 162)

[Borgo et al., 2019b] Borgo, S., Cesta, A., Orlandini, A., and Umbrico, A. (2019b). Knowledge-based adaptive agents for manufacturing domains. *Engineering with Computers*, 35(3):755–779. (Cited on p. 112)

[Borgo et al., 2021] Borgo, S., Ferrario, R., Gangemi, A., Guarino, N., Masolo, C., Porello, D., Sanfilippo, E. M., and Vieu, L. (2021). Dolce: A descriptive ontology for linguistic and cognitive engineering. *Applied Ontology*, Preprint:1–25. Preprint. (Cited on p. 19, 87, 115, 131, 161)

[Borgo et al., 2014] Borgo, S., Franssen, M., Garbacz, P., Kitamura, Y., Mizoguchi, R., and Vermaas, P. E. (2014). Technical artifacts: An integrated perspective. *Applied Ontology*, 9:217–235. 3-4. (Cited on p. 161)

[Borst et al., 1997] Borst, P., Akkermans, H., and Top, J. (1997). Engineering ontologies. *International Journal of Human-Computer Studies*, 46:365–406. (Cited on p. 2)

[Bozcuoğlu et al., 2019] Bozcuoğlu, A. K., Furuta, Y., Okada, K., Beetz, M., and Inaba, M. (2019). Continuous modeling of affordances in a symbolic knowledge base. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5452–5458. (Cited on p. 112)

[Brooks, 1991] Brooks, R. A. (1991). Intelligence without representation. *Artificial Intelligence*, 47(1):139–159. (Cited on p. 163)

[Bruno et al., 2017] Bruno, B., Chong, N. Y., Kamide, H., Kanoria, S., Lee, J., Lim, Y., Pandey, A. K., Papadopoulos, C., Papadopoulos, I., Pecora, F., Saffiotti, A., and Sgorbissa, A. (2017). Paving the way for culturally competent robots: A position paper. In *26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 553–560. (Cited on p. 176)

[Bruno et al., 2019a] Bruno, B., Chong, N. Y., Kamide, H., Kanoria, S., Lee, J., Lim, Y., Pandey, A. K., Papadopoulos, C., Papadopoulos, I., Pecora, F., Saffiotti, A., and Sgorbissa, A. (2019a).

The caresses eu-japan project: Making assistive robots culturally competent. In Casiddu, N., Porfirione, C., Monteriù, A., and Cavallo, F., editors, *2017 Italian Forum of Ambient Assisted Living*, pages 151–169. Springer International Publishing. (Cited on p. 86, 112, 170, 176)

[Bruno et al., 2018] Bruno, B., Menicatti, R., Recchiuto, C. T., Lagrue, E., Pandey, A. K., and Sgorbissa, A. (2018). Culturally-competent human-robot verbal interaction. In *15th International Conference on Ubiquitous Robots (UR)*, pages 388–395. (Cited on p. 29, 33)

[Bruno et al., 2019b] Bruno, B., Recchiuto, C. T., Papadopoulos, I., Saffiotti, A., Koulouglioti, C., Menicatti, R., Mastrogiovanni, F., Zaccaria, R., and Sgorbissa, A. (2019b). Knowledge representation for culturally competent personal robots: Requirements, design principles, implementation, and assessment. *International Journal of Social Robotics*, 11(3):515–538. (Cited on p. 29, 30, 32, 33)

[Buehler and Pagnucco, 2014] Buehler, J. and Pagnucco, M. (2014). A framework for task planning in heterogeneous multi robot systems based on robot capabilities. *Proceedings of the AAAI Conference on Artificial Intelligence*, 28(1). (Cited on p. 164)

[Burkart and Huber, 2021] Burkart, N. and Huber, M. F. (2021). A survey on the explainability of supervised machine learning. *Journal of Artificial Intelligence Research*, 70:245–317. (Cited on p. 5)

[Byner et al., 2019] Byner, C., Matthias, B., and Ding, H. (2019). Dynamic speed and separation monitoring for collaborative robot applications – concepts and performance. *Robotics and Computer-Integrated Manufacturing*, 58:239 – 252. (Cited on p. 66)

[Campomaggiore et al., 2019] Campomaggiore, A., Costanzo, M., Lettera, G., and Natale, C. (2019). A fuzzy inference approach to control robot speed in human-robot shared workspaces. In *16th International Conference on Informatics in Control, Automation and Robotics, ICINCO 2019*, volume 2, pages 78–87. SciTePress. (Cited on p. 64, 66, 69)

[Canal et al., 2022] Canal, G., Krivić, S., Luff, P., and Coles, A. (2022). PlanVerb: Domain-Independent Verbalization and Summary of Task Plans. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36. (Cited on p. 112, 130)

[Carey, 2018] Carey, P. (2018). *Data protection: a practical guide to UK and EU law*. Oxford University Press, Inc. (Cited on p. 4)

[Carr, 2008] Carr, D. (2008). Narrative explanation and its malcontents. *History and Theory*, 47(1):19–30. (Cited on p. 110, 111)

[Cashmore et al., 2015] Cashmore, M., Fox, M., Long, D., Magazzeni, D., Ridder, B., Carrera, A., Palomeras, N., Hurtos, N., and Carreras, M. (2015). Rosplan: Planning in the robot operating system. In *Proceedings of the international conference on automated planning and scheduling*, volume 25, pages 333–341. (Cited on p. 135)

[Celiktutan et al., 2019] Celiktutan, O., Skordos, E., and Gunes, H. (2019). Multimodal human-human-robot interactions (mhhri) dataset for studying personality and engagement. *IEEE Transactions on Affective Computing*, 10(4):484–497. (Cited on p. 41)

[Chacón et al., 2020] Chacón, A., Angulo, C., and Ponsa, P. (2020). *New Trends in the Use of Artificial Intelligence for the Industry 4.0*, chapter Developing cognitive advisor agents for operators in industry 4.0, pages 127–142. IntechOpen. (Cited on p. 96)

[Chakraborti et al., 2020] Chakraborti, T., Sreedharan, S., and Kambhampati, S. (2020). The emerging landscape of explainable automated planning & decision making. In Bessiere, C., editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 4803–4811. International Joint Conferences on Artificial Intelligence Organization. Survey track. (Cited on p. 5, 110)

[Chall and Dale, 1995] Chall, J. and Dale, E. (1995). *Readability Revisited: The New Dale-Chall Readability Formula*. Brookline Books. (Cited on p. 148)

[Chandrasekaran et al., 1999] Chandrasekaran, B., Josephson, J., and Benjamins, V. (1999). What are ontologies, and why do we need them? *IEEE Intelligent Systems and their Applications*, 14(1):20–26. (Cited on p. 17)

[Chari et al., 2020] Chari, S., Seneviratne, O., Gruen, D. M., Foreman, M. A., Das, A. K., and McGuinness, D. L. (2020). Explanation ontology: A model of explanations for user-centered ai. In Pan, J. Z., Tamma, V., d'Amato, C., Janowicz, K., Fu, B., Polleres, A., Seneviratne, O., and Kagal, L., editors, *The Semantic Web – ISWC 2020*, pages 228–243, Cham. Springer International Publishing. (Cited on p. 6)

[Chella et al., 2002] Chella, A., Cossentino, M., Pirrone, R., and Ruisi, A. (2002). Modeling ontologies for robotic environments. In *Proceedings of the 14th International Conference on Software Engineering and Knowledge Engineering*, page 77–80. Association for Computing Machinery. (Cited on p. 162)

[Chen et al., 2021] Chen, W. H., Foo, G., Kara, S., and Pagnucco, M. (2021). Automated generation and execution of disassembly actions. *Robotics and Computer-Integrated Manufacturing*, 68:102056. (Cited on p. 86, 112)

[Cherubini et al., 2016] Cherubini, A., Passama, R., Crosnier, A., Lasnier, A., and Fraisse, P. (2016). Collaborative manufacturing with physical human–robot interaction. *Robotics and Computer-Integrated Manufacturing*, 40:1–13. (Cited on p. 42)

[Clocksin and Mellish, 2012] Clocksin, W. F. and Mellish, C. S. (2012). *Programming in Prolog: Using the ISO standard*. Springer-Verlag Berlin Heidelberg. (Cited on p. 137)

[Compton et al., 2012] Compton, M., Barnaghi, P., Bermudez, L., García-Castro, R., Corcho, O., Cox, S., Graybeal, J., Hauswirth, M., Henson, C., Herzog, A., Huang, V., Janowicz, K., Kelsey, W. D., Le Phuoc, D., Lefort, L., Leggieri, M., Neuhaus, H., Nikolov, A., Page, K., Passant, A., Sheth, A., and Taylor, K. (2012). The ssn ontology of the w3c semantic sensor network incubator group. *Journal of Web Semantics*, 17:25–32. (Cited on p. 165)

[Dalianis and Hovy, 1996] Dalianis, H. and Hovy, E. (1996). Aggregation in natural language generation. In Adorni, G. and Zock, M., editors, *Trends in Natural Language Generation An Artificial Intelligence Perspective*, pages 88–105. Springer. (Cited on p. 112, 117, 144)

[Davidson, 2001] Davidson, D. (2001). *Essays on Actions and Events*. Oxford University Press. (Cited on p. 163)

[de Gea Fernández et al., 2017] de Gea Fernández, J., Mronga, D., Günther, M., Wirkus, M., Schröer, M., Stiene, S., Kirchner, E., Bargsten, V., Bänziger, T., Teiwes, J., et al. (2017). imrk: Demonstrator for intelligent and intuitive human–robot collaboration in industrial manufacturing. *KI-Künstliche Intelligenz*, 31(2):203–207. (Cited on p. 42)

[Di Paolo, 2005] Di Paolo, E. A. (2005). Autopoiesis, adaptivity, teleology, agency. *Phenomenology and the Cognitive Sciences*, 4(4):429–452. (Cited on p. 93)

[Diab et al., 2019] Diab, M., Akbari, A., Ud Din, M., and Rosell, J. (2019). Pmk—a knowledge processing framework for autonomous robotics perception and manipulation. *Sensors*, 19(5). (Cited on p. 29, 30, 31, 32, 86, 171, 175)

[Diab et al., 2018] Diab, M., Muhayyuddin, Akbari, A., and Rosell, J. (2018). An ontology framework for physics-based manipulation planning. In Ollero, A., Sanfeliu, A., Montano, L., Lau, N., and Cardeira, C., editors, *ROBOT 2017: Third Iberian Robotics Conference*, pages 452–464. Springer International Publishing. (Cited on p. 29, 30)

[Dillenbourg, 1999] Dillenbourg, P. (1999). *Collaborative learning: Cognitive and computational approaches. (Advances in learning and instruction series)*, chapter 1 - What do you mean by collaborative learning?, pages 1–19. Elsevier. (Cited on p. 89, 90)

[Dix, 2009] Dix, A. (2009). *Human-computer interaction*. Springer. (Cited on p. 165)

[Dogmus et al., 2015] Dogmus, Z., Erdem, E., and Patoglu, V. (2015). Rehabrobo-onto: Design, development and maintenance of a rehabilitation robotics ontology on the cloud. *Robotics and Computer-Integrated Manufacturing*, 33:100–109. (Cited on p. 178)

[Dogmus et al., 2019] Dogmus, Z., Erdem, E., and Patoglu, V. (2019). Rehabrobo-query: Answering natural language queries about rehabilitation robotics ontology on the cloud. *Semantic Web*, 10:605–629. 3. (Cited on p. 170, 178)

[Eguíluz et al., 2020] Eguíluz, A., Rodríguez-Gómez, J., Martínez-de Dios, J., and Ollero, A. (2020). Asynchronous event-based line tracking for time-to-contact maneuvers in uas. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5978–5985. (Cited on p. 66)

[Elkan and Greiner, 1993] Elkan, C. and Greiner, R. (1993). Building large knowledge-based systems: Representation and inference in the cyc project: D.b. lenat and r.v. guha. *Artificial Intelligence*, 61(1):41–52. (Cited on p. 19)

[Fazel-Zarandi and Fox, 2013] Fazel-Zarandi, M. and Fox, M. S. (2013). Inferring and validating skills and competencies over time. *Applied Ontoloy*, 8(3):131–177. (Cited on p. 164)

[Fernández-López et al., 1997] Fernández-López, M., Gómez-Pérez, A., and Juristo, N. (1997). Methontology: from ontological art towards ontological engineering. Technical Report SS-97-06, AAAI Press, Menlo Park, California. p. 34–40. (Cited on p. 87, 130)

[Fiorini et al., 2017] Fiorini, S. R., Bermejo-Alonso, J., Gonçalves, P., Pignaton de Freitas, E., Olivares-Alarcos, A., Olszewska, J. I., Prestes, E., Schlenoff, C., Ragavan, S. V., Redfield, S., Spencer, B., and Li, H. (2017). A suite of ontologies for robotics and automation [industrial activities]. *IEEE Robotics & Automation Magazine*, 24(1):8–11. (Cited on p. 85, 112, 129)

[Flores et al., 2018] Flores, J. G., Meza, I., Colin, É., Gardent, C., Gangemi, A., and Pineda, L. A. (2018). Robot experience stories: First person generation of robotic task narratives in sitlog. *Journal of Intelligent & Fuzzy Systems*, 34:3291–3300. 5. (Cited on p. 112, 130)

[Fox and Long, 2003] Fox, M. and Long, D. (2003). The 3rd international planning competition: Results and analysis. *Journal of Artificial Intelligence Research*, 20:1–59. (Cited on p. 147)

[Gangemi et al., 2004] Gangemi, A., Borgo, S., Catenacci, C., and Lehmann, J. (2004). Task taxonomies for knowledge content d07. Technical report, Metokis Project. (Cited on p. 163, 165)

[Gangemi et al., 2003] Gangemi, A., Guarino, N., Masolo, C., and Oltramari, A. (2003). Sweetening wordnet with dolce. *AI Magazine*, 24(3):13. (Cited on p. 134)

[Gangemi and Mika, 2003] Gangemi, A. and Mika, P. (2003). Understanding the semantic web through descriptions and situations. In Meersman, R., Tari, Z., and Schmidt, D. C., editors, *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE*, pages 689–706. Springer. (Cited on p. 166)

[Garcia et al., 2016] Garcia, A. J. S., Figueroa, H. V. R., Hernandez, A. M., Verdin, M. K. C., and Vega, G. C. (2016). Estimation of time-to-contact from tau-margin and statistical analysis of behavior. In *2016 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pages 1–6. (Cited on p. 67)

[Garcia-Camacho et al., 2020] Garcia-Camacho, I., Lippi, M., Welle, M. C., Yin, H., Antonova, R., Varava, A., Borras, J., Torras, C., Marino, A., Alenyà, G., and Kragic, D. (2020). Benchmarking bimanual cloth manipulation. *IEEE Robotics and Automation Letters*, 5(2):1111–1118. (Cited on p. 97)

[Gassó Loncan Vallecillo et al., 2020] Gassó Loncan Vallecillo, J., Olivares-Alarcos, A., and Alenyà, G. (2020). Visual feedback for humans about robots' perception in collaborative environments. Technical Report IRI-TR-20-03, Institut de Robòtica i Informàtica Industrial, CSIC-UPC. (Cited on p. 8)

[Gaz et al., 2018] Gaz, C., Magrini, E., and De Luca, A. (2018). A model-based residual approach for human-robot collaboration during manual polishing operations. *Mechatronics*, 55:234–247. (Cited on p. 43)

[Gennari et al., 2003] Gennari, J. H., Musen, M. A., Fergerson, R. W., Grosso, W. E., Crubézy, M., Eriksson, H., Noy, N. F., and Tu, S. W. (2003). The evolution of protégé: an environment for knowledge-based systems development. *International Journal of Human-Computer Studies*, 58(1):89 – 123. (Cited on p. 98)

[Georgara et al., 2022] Georgara, A., Rodriguez Aguilar, J. A., and Sierra, C. (2022). Building contrastive explanations for multi-agent team formation. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '22, page 516–524, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems. (Cited on p. 148)

[Gervasi et al., 2020] Gervasi, R., Mastrogiacomo, L., and Franceschini, F. (2020). A conceptual framework to evaluate human-robot collaboration. *The International Journal of Advanced Manufacturing Technology*, 108(3):841–865. (Cited on p. 84)

[Gibson, 1979] Gibson, J. J. (1979). *The Ecological Approach to Visual Perception*. Houghton Mifflin. (Cited on p. 162)

[Gil, 2005] Gil, Y. (2005). Description logics and planning. *AI Magazine*, 26(2):73–84. (Cited on p. 164)

[Gjorven et al., 2006] Gjorven, E., Eliassen, F., and Aagedal, J. (2006). Quality of adaptation. In *IEEE International Conference on Autonomic and Autonomous Systems*, pages 9–9. (Cited on p. 92, 93)

[Glimm et al., 2014] Glimm, B., Horrocks, I., Motik, B., Stoilos, G., and Wang, Z. (2014). Hermit: An owl 2 reasoner. *Journal of Automated Reasoning*, 53(3):245–269. (Cited on p. 99)

[Gómez-Pérez et al., 2004] Gómez-Pérez, A., Fernández-López, M., and Corcho, O. (2004). *Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web*. Advanced Information and Knowledge Processing. Springer, 1st edition. (Cited on p. 19)

[Gonçalves and Torres, 2015] Gonçalves, P. J. and Torres, P. M. (2015). Knowledge representation applied to robotic orthopedic surgery. *Robotics and Computer-Integrated Manufacturing*, 33:90–99. (Cited on p. 29, 32, 171, 174)

[Gonçalves et al., 2021a] Gonçalves, P. J., Olivares-Alarcos, A., Bermejo-Alonso, J., Borgo, S., Diab, M., Habib, M., Nakawala, H., Ragavan, S. V., Sanz, R., Tosello, E., and Li, H. (2021a). Ieee standard for autonomous robotics ontology [standards]. *IEEE Robotics & Automation Magazine*, 28(3):171–173. (Cited on p. 8)

[Gonçalves et al., 2021b] Gonçalves, P. J., Olivares-Alarcos, A., Bermejo-Alonso, J., Borgo, S., Diab, M., Habib, M., Nakawala, H., Ragavan, S. V., Sanz, R., Tosello, E., and Li, H. (2021b). Ieee standard for autonomous robotics ontology [standards]. *IEEE Robotics & Automation Magazine*, 28(3):171–173. (Cited on p. 112, 129)

[Gopinath et al., 2021] Gopinath, V., Johansen, K., Derelöv, M., Åke Gustafsson, and Axelsson, S. (2021). Safe collaborative assembly on a continuously moving line with large industrial robots. *Robotics and Computer-Integrated Manufacturing*, 67:102048. (Cited on p. 84)

[Grice, 1975] Grice, H. P. (1975). Logic and conversation. In Cole, P. and Morgan, J. L., editors, *Syntax and Semantics: Vol. 3: Speech Acts*, pages 41–58. Academic Press, New York. (Cited on p. 121)

[Gruber, 1993] Gruber, T. (1993). A translation approach to portable ontologies. *Knowledge Acquisition*, 5(2):199–220. (Cited on p. 2)

[Gualtieri et al., 2021] Gualtieri, L., Rauch, E., and Vidoni, R. (2021). Emerging research fields in safety and ergonomics in industrial collaborative robotics: A systematic literature review. *Robotics and Computer-Integrated Manufacturing*, 67:101998. (Cited on p. 84)

[Guarino, 1998] Guarino, N. (1998). Formal ontology in information systems. In *Proceedings of the International Conference on Formal Ontology in Information Systems (FOIS)*, pages 3–15. IOS Press. (Cited on p. 19)

[Guarino and Giaretta, 1995] Guarino, N. and Giaretta, P. (1995). Ontologies and knowledge bases: Towards a terminological clarification. In N, M., editor, *Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing (KBKS'95)*, pages 25–32. IOS Press. (Cited on p. 2)

[Guarino et al., 2009] Guarino, N., Oberle, D., and Staab, S. (2009). What is an ontology? In Staab, S. and Studer, R., editors, *Handbook on Ontologies*, pages 1–17. Springer, 2 edition. (Cited on p. 2, 17)

[Guizzardi and Guarino, 2023] Guizzardi, G. and Guarino, N. (2023). Semantics, ontology and explanation. (Cited on p. 5)

[Gunning and Aha, 2019] Gunning, D. and Aha, D. (2019). Darpa's explainable artificial intelligence (xai) program. *AI Magazine*, 40(2):44–58. (Cited on p. 4)

[Haage et al., 2011] Haage, M., Malec, J., Nilsson, A., Nilsson, K., and Nowaczyk, S. (2011). Declarative-knowledge-based reconfiguration of automation systems using a blackboard architecture. In *Proceedings of the Eleventh Scandinavian Conference on Artificial Intelligence*, volume 227, pages 163–172. IOS Press. (Cited on p. 173)

[Haidu et al., 2018] Haidu, A., Beßler, D., Bozcuoğlu, A. K., and Beetz, M. (2018). Knowrobsim — game engine-enabled knowledge processing towards cognition-enabled robot control. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4491–4498. (Cited on p. 172)

[Halsey et al., 2004] Halsey, K., Long, D., and Fox, M. (2004). Crikey-a temporal planner looking at the integration of scheduling and planning. In *Workshop on Integrating Planning into Scheduling, ICAPS*, pages 46–52. Citeseer. (Cited on p. 147)

[Harnad, 1990] Harnad, S. (1990). The symbol grounding problem. *Physica D: Nonlinear Phenomena*, 42(1):335–346. (Cited on p. 3)

[Hecht and Savelsbergh, 2004] Hecht, H. and Savelsbergh, G. (2004). *Time-to-contact*, volume 135. Elsevier. (Cited on p. 64)

[Hesslow, 2012] Hesslow, G. (2012). The current status of the simulation theory of cognition. *Brain Research*, 1428:71–79. (Cited on p. 172)

[Hou et al., 2014] Hou, J., List, G. F., and Guo, X. (2014). New algorithms for computing the time-to-collision in freeway traffic simulation models. *Computational Intelligence and Neuroscience*, 2014:761047. (Cited on p. 67)

[Huang and Sun, 2018] Huang, Y. and Sun, Y. (2018). A dataset of daily interactive manipulation. (Cited on p. 42)

[IEEE-SA, 2021] IEEE-SA (2021). IEEE Standard for Autonomous Robotics (AuR) Ontology. *IEEE Std. 1872.2-2021*, pages 1–49. (Cited on p. 8)

[ISO 10218-1:2011, 2011] ISO 10218-1:2011 (2011). 10218 Robots and robotic devices – Safety requirements for industrial robots – Part 1: Robots. Standard, International Organization for Standardization. (Cited on p. 64, 65, 68, 88, 95)

[ISO 10218-2:2011, 2011] ISO 10218-2:2011 (2011). 10218 Robots and robotic devices – Safety requirements for industrial robots – Part 2: Robot systems and integration. Standard, International Organization for Standardization. (Cited on p. 64, 65, 84, 88, 90, 95)

[ISO 12100:2010, 2010] ISO 12100:2010 (2010). 12100 Safety of machinery — General principles for design — Risk assessment and risk reduction. Standard, International Organization for Standardization. (Cited on p. 95, 96)

[ISO 8373:2021, 2021] ISO 8373:2021 (2021). Robotics – Vocabulary. Standard, International Organization for Standardization. (Cited on p. 20, 22, 169)

[ISO/TS 15066:2016, 2016] ISO/TS 15066:2016 (2016). TS 15066 Robots and robotic devices – Collaborative robots. Standard, International Organization for Standardization. (Cited on p. 64, 65, 69)

[Jacobsson et al., 2016] Jacobsson, L., Malec, J., and Nilsson, K. (2016). Modularization of skill ontologies for industrial robots. In *Proceedings of ISR 2016: 47st International Symposium on Robotics*, pages 1–6. (Cited on p. 173)

[Järvenpää et al., 2016] Järvenpää, E., Lanz, M., Tuokko, R., et al. (2016). Application of a capability-based adaptation methodology to a small-size production system. *International Journal of Manufacturing Technology and Management*, 30(1/2):67–86. (Cited on p. 92, 93)

[Jayagopi et al., 2013] Jayagopi, D. B., Sheiki, S., Klotz, D., Wienke, J., Odobez, J.-M., Wrede, S., Khalidov, V., Nyugen, L., Wrede, B., and Gatica-Perez, D. (2013). The vernissage corpus: A conversational human-robot-interaction dataset. In *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 149–150. (Cited on p. 41)

[Jorge et al., 2015] Jorge, V. A., Rey, V. F., Maffei, R., Fiorini, S. R., Carbonera, J. L., Branchi, F., Meireles, J. P., Franco, G. S., Farina, F., da Silva, T. S., Kolberg, M., Abel, M., and Prestes, E. (2015). Exploring the ieee ontology for robotics and automation for heterogeneous agent interaction. *Robotics and Computer-Integrated Manufacturing*, 33:12–20. (Cited on p. 174)

[Joseph et al., 2020] Joseph, L., Pickard, J. K., Padois, V., and Daney, D. (2020). Online velocity constraint adaptation for safe and efficient human-robot workspace sharing. In *2020*

IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 11045–11051. (Cited on p. 66)

[Kaneta et al., 2010] Kaneta, Y., Hagisaka, Y., and Ito, K. (2010). Determination of time to contact and application to timing control of mobile robot. In *2010 IEEE International Conference on Robotics and Biomimetics*, pages 161–166. (Cited on p. 66)

[Kaneta et al., 2010] Kaneta, Y., Hagisaka, Y., and Ito, K. (2010). Determination of time to contact and application to timing control of mobile robot. In *2010 IEEE international conference on robotics and biomimetics*, pages 161–166. (Cited on p. 67)

[Karray et al., 2019] Karray, M. H., Ameri, F., Hodkiewicz, M., and Louge, T. (2019). Romain: Towards a bfo compliant reference ontology for industrial maintenance. *Applied Ontology*, 14:155–177. 2. (Cited on p. 85, 86)

[Kendoul, 2014] Kendoul, F. (2014). Four-dimensional guidance and control of movement using time-to-contact: Application to automated docking and landing of unmanned rotorcraft systems. *The International Journal of Robotics Research*, 33(2):237–267. (Cited on p. 66)

[Khaliq et al., 2018] Khaliq, A. A., Köckemann, U., Pecora, F., Saffiotti, A., Bruno, B., Recchiuto, C. T., Sgorbissa, A., Bui, H.-D., and Chong, N. Y. (2018). Culturally aware planning and execution of robot actions. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 326–332. (Cited on p. 32)

[Kim et al., 2021] Kim, W., Peternel, L., Lorenzini, M., Babič, J., and Ajoudani, A. (2021). A human-robot collaboration framework for improving ergonomics during dexterous operation of power tools. *Robotics and Computer-Integrated Manufacturing*, 68:102084. (Cited on p. 84)

[Kolfschoten, 2007] Kolfschoten, G. L. (2007). *Theoretical foundations for collaboration engineering*. PhD thesis, Faculty of Technology Policy and Management. Delft University of Technology., Jaffalaan 5, 2628 BX Delft, the Netherlands. (Cited on p. 89, 90)

[Krarup et al., 2021] Krarup, B., Krivic, S., Magazzeni, D., Long, D., Cashmore, M., and Smith, D. E. (2021). Contrastive explanations of plans through model restrictions. *Journal of Artificial Intelligence Research*, 72:533–612. (Cited on p. 139)

[Krüger et al., 2007] Krüger, V., Kragic, D., Ude, A., and Geib, C. (2007). The meaning of action: a review on action recognition and mapping. *Advanced Robotics*, 21(13):1473–1501. (Cited on p. 163)

[Kunze et al., 2011] Kunze, L., Roehm, T., and Beetz, M. (2011). Towards semantic robot description languages. In *2011 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5589–5595. (Cited on p. 164)

[Labov and Waletzky, 1997] Labov, W. and Waletzky, J. (1997). Narrative analysis: Oral versions of personal experience1. *Journal of Narrative and Life History*, 7(1-4):3–38. (Cited on p. 111)

[Langley et al., 2009] Langley, P., Laird, J. E., and Rogers, S. (2009). Cognitive architectures: Research issues and challenges. *Cognitive Systems Research*, 10(2):141–160. (Cited on p. 20, 22, 166)

[Langley et al., 2017] Langley, P., Meadows, B., Sridharan, M., and Choi, D. (2017). Explainable agency for intelligent autonomous systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, page 4762–4763. AAAI Press. (Cited on p. 5, 6, 110, 129, 130)

[Lasota et al., 2014] Lasota, P. A., Rossano, G. F., and Shah, J. A. (2014). Toward safe close-proximity human-robot interaction with standard industrial robots. In *2014 IEEE International Conference on Automation Science and Engineering (CASE)*, pages 339–344. (Cited on p. 66)

[Lawrence, 2003] Lawrence, N. D. (2003). Gaussian process latent variable models for visualisation of high dimensional data. In *Proceedings of the 16th International Conference on Neural Information Processing Systems (NeurIPS)*, page 329–336, Cambridge, MA, USA. (Cited on p. 49)

[Lee et al., 2002] Lee, Y. W., Strong, D. M., Kahn, B. K., and Wang, R. Y. (2002). Aimq: a methodology for information quality assessment. *Information & Management*, 40(2):133–146. (Cited on p. 122, 179)

[Lemaignan et al., 2011] Lemaignan, S., Ros, R., Alami, R., and Beetz, M. (2011). What are you talking about? grounding dialogue in a perspective-aware robotic architecture. In *20th IEEE International Symposium in Robot and Human Interactive Communication (RO-MAN)*, pages 107–112. (Cited on p. 29, 33)

[Lemaignan et al., 2010] Lemaignan, S., Ros, R., Mösenlechner, L., Alami, R., and Beetz, M. (2010). Oro, a knowledge management platform for cognitive architectures in robotics. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3548–3553. (Cited on p. 86, 170, 175, 176)

[Levine and Williams, 2014] Levine, S. J. and Williams, B. C. (2014). Concurrent plan recognition and execution for human-robot teams. In *Proceedings of the Twenty-Fourth International Conference on Automated Planning and Scheduling*, page 490–498. AAAI Press. (Cited on p. 85)

[Levine and Williams, 2018] Levine, S. J. and Williams, B. C. (2018). Watching and acting together: Concurrent plan recognition and adaptation for human-robot teams. *Journal of Artificial Intelligence Research*, 63:281–359. (Cited on p. 85)

[Li et al., 2016] Li, Y., Zhang, L., and Song, Y. (2016). A vehicular collision warning algorithm based on the time-to-collision estimation under connected environment. In *14th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pages 1–4. (Cited on p. 66)

[Liang, 2018] Liang, J. S. (2018). An ontology-oriented knowledge methodology for process planning in additive layer manufacturing. *Robotics and Computer-Integrated Manufacturing*, 53:28 – 44. (Cited on p. 86)

[Liang, 2020] Liang, J. S. (2020). A process-based automotive troubleshooting service and knowledge management system in collaborative environment. *Robotics and Computer-Integrated Manufacturing*, 61:101836. (Cited on p. 86)

[Lim, 2019] Lim, G. H. (2019). Shared representations of actions for alternative suggestion with incomplete information. *Robotics and Autonomous Systems*, 116:38–50. (Cited on p. 178)

[Lim et al., 2011] Lim, G. H., Suh, I. H., and Suh, H. (2011). Ontology-based unified robot knowledge for service robots in indoor environments. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 41(3):492–509. (Cited on p. 170, 175, 178)

[Lints, 2012] Lints, T. (2012). The essentials of defining adaptation. *IEEE Aerospace and Electronic Systems Magazine*, 27(1):37–41. (Cited on p. 92, 93)

[Liu and Nocedal, 1989] Liu, D. C. and Nocedal, J. (1989). On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1):503–528. (Cited on p. 49)

[Liu and Wang, 2021] Liu, H. and Wang, L. (2021). Collision-free human-robot collaboration based on context awareness. *Robotics and Computer-Integrated Manufacturing*, 67:101997. (Cited on p. 84)

[Losey et al., 2018] Losey, D. P., McDonald, C. G., Battaglia, E., and O'Malley, M. K. (2018). A Review of Intent Detection, Arbitration, and Communication Aspects of Shared Control for Physical Human–Robot Interaction. *Applied Mechanics Reviews*, 70(1):010804. (Cited on p. 43)

[Maceira et al., 2020] Maceira, M., Olivares-Alarcos, A., and Alenyà, G. (2020). Recurrent neural networks for inferring intentions in shared tasks for industrial collaborative robots. In *2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, pages 665–670. (Cited on p. 8)

[Magistris et al., 2018] Magistris, G. D., Munawar, A., Pham, T.-H., Inoue, T., Vinayavekhin, P., and Tachibana, R. (2018). Experimental force-torque dataset for robot learning of multi-shape insertion. (Cited on p. 41)

[Magrini et al., 2020] Magrini, E., Ferraguti, F., Ronga, A. J., Pini, F., Luca, A. D., and Leali, F. (2020). Human-robot coexistence and interaction in open industrial cells. *Robotics and Computer-Integrated Manufacturing*, 61:101846. (Cited on p. 64, 65)

[Marconi et al., 2012] Marconi, L., Melchiorri, C., Beetz, M., Pangercic, D., Siegwart, R., Leutenegger, S., Carloni, R., Stramigioli, S., Bruyninckx, H., Doherty, P., Kleiner, A., Lippiello, V., Finzi, A., Siciliano, B., Sala, A., and Tomatis, N. (2012). The sherpa project: Smart collaboration between humans and ground-aerial robots for improving rescuing activities in alpine environments. In *2012 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 1–4. (Cited on p. 172)

[Martín H. et al., 2008] Martín H., J. A., de Lope, J., and Maravall, D. (2008). Adaptation, anticipation and rationality in natural and artificial systems: computational paradigms mimicking nature. *Natural Computing*, 8(4):757. (Cited on p. 92, 93)

[Marvel and Norcross, 2017] Marvel, J. A. and Norcross, R. (2017). Implementing speed and separation monitoring in collaborative robot workcells. *Robotics and Computer-Integrated Manufacturing*, 44:144–155. (Cited on p. 64, 69)

[Masolo and Borgo, 2005] Masolo, C. and Borgo, S. (2005). Qualities in formal ontology. In *Workshop on Foundational Aspects of Ontologies (FOnt 2005) at KI 2005: Advances in Artificial Intelligence*, pages 2–16. (Cited on p. 162)

[Maurtua et al., 2017] Maurtua, I., Ibarguren, A., Kildal, J., Susperregi, L., and Sierra, B. (2017). Human–robot collaboration in industrial applications: Safety, interaction and trust. *International Journal of Advanced Robotic Systems*, 14(4):1729881417716010. (Cited on p. 42)

[Mazhar et al., 2018] Mazhar, O., Ramdani, S., Navarro, B., Passama, R., and Cherubini, A. (2018). Towards real-time physical human-robot interaction using skeleton information and hand gestures. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–6. (Cited on p. 43)

[McBride, 2004] McBride, B. (2004). *The Resource Description Framework (RDF) and its Vocabulary Description Language RDFS*, pages 51–65. Springer Berlin Heidelberg, Berlin, Heidelberg. (Cited on p. 137)

[McDermott et al., 1998] McDermott, D., Ghallab, M., Howe, A., Knoblock, C., Ram, A., Veloso, M., Weld, D., and Wilkins, D. (1998). PDDL—The Planning Domain Definition Language. Technical Report TR98003/DCS TR1165., New Haven, CT: Yale Center for Computational Vision and Control. (Cited on p. 164)

[Melchiorre et al., 2021] Melchiorre, M., Scimmi, L. S., Mauro, S., and Pastorelli, S. P. (2021). Vision-based control architecture for human–robot hand-over applications. *Asian Journal of Control*, 23(1):105–117. (Cited on p. 95)

[Menicatti et al., 2017] Menicatti, R., Bruno, B., and Sgorbissa, A. (2017). Modelling the influence of cultural information on vision-based human home activity recognition. In *14th IEEE International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, pages 32–38. (Cited on p. 28, 29)

[Michalos et al., 2014] Michalos, G., Makris, S., Spiliotopoulos, J., Misios, I., Tsarouchi, P., and Chryssolouris, G. (2014). Robo-partner: Seamless human-robot cooperation for intelligent, flexible and safe operations in the assembly factories of the future. *Procedia CIRP*, 23:71–76. (Cited on p. 40)

[Michalos et al., 2015] Michalos, G., Makris, S., Tsarouchi, P., Guasch, T., Kontovrakis, D., and Chryssolouris, G. (2015). Design considerations for safe human-robot collaborative workplaces. *Procedia CIRP*, 37:248–253. (Cited on p. 40)

[Miller, 2019] Miller, T. (2019). Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38. (Cited on p. 139)

[Mizoguchi et al., 2016] Mizoguchi, R., Kitamura, Y., and Borgo, S. (2016). A unifying definition for artifact and biological functions. *Applied Ontology*, 11(2):129–154. (Cited on p. 163)

[Mohammad et al., 2008] Mohammad, Y., Xu, Y., Matsumura, K., and Nishida, T. (2008). The h3r explanation corpus human-human and base human-robot interaction dataset. In *2008 International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pages 201–206. (Cited on p. 41)

[Mohd Ali et al., 2019] Mohd Ali, M., Rai, R., Otte, J. N., and Smith, B. (2019). A product life cycle ontology for additive manufacturing. *Computers in Industry*, 105:191 – 203. (Cited on p. 85, 86)

[Moralez, 2023] Moralez, L. A. (2023). Affordance ontology: towards a unified description of affordances as events. *International Journal of Undergraduate Research and Creative Activities*, 8(2):4. (Cited on p. 162)

[Munzer et al., 2018] Munzer, T., Toussaint, M., and Lopes, M. (2018). Efficient behavior learning in human–robot collaboration. *Autonomous Robots*, 42(5):1103–1115. (Cited on p. 43)

[Neuhaus et al., 2004] Neuhaus, F., Grenon, P., and Smith, B. (2004). A formal theory of substances, qualities, and universals. In *Proceedings of the International Conference on Formal Ontology in Information Systems (FOIS)*. IOS Press. (Cited on p. 162)

[Ngonga Ngomo et al., 2019] Ngonga Ngomo, A.-C., Moussallem, D., and Bühmann, L. (2019). A holistic natural language generation framework for the semantic web. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 819–828, Varna, Bulgaria. INCOMA Ltd. (Cited on p. 112)

[Nguyen et al., 2019] Nguyen, V., Son, T. C., and Pontelli, E. (2019). Natural language generation from ontologies. In Alferes, J. J. and Johansson, M., editors, *Practical Aspects of Declarative Languages*, pages 64–81, Cham. Springer. (Cited on p. 112)

[Nikolaidis et al., 2018] Nikolaidis, S., Kwon, M., Forlizzi, J., and Srinivasa, S. (2018). Planning with verbal communication for human-robot collaboration. *J. Hum.-Robot Interact.*, 7(3). (Cited on p. 96)

[Nikolakis et al., 2019] Nikolakis, N., Maratos, V., and Makris, S. (2019). A cyber physical system (cps) approach for safe human-robot collaboration in a shared workplace. *Robotics and Computer-Integrated Manufacturing*, 56:233 – 243. (Cited on p. 64, 65)

[Niles and Pease, 2001] Niles, I. and Pease, A. (2001). Towards a standard upper ontology. In *Proceedings of the International Conference on Formal Ontology in Information Systems (FOIS)*, page 2–9. Association for Computing Machinery. (Cited on p. 19, 161, 173, 175)

[Norman, 2002] Norman, D. A. (2002). *The Design of Everyday Things*. Basic Books, Inc. (Cited on p. 162)

[O'Connor and Das, 2009] O'Connor, M. and Das, A. (2009). Sqwrl: A query language for owl. In *Proceedings of the 6th International Conference on OWL: Experiences and Directions - Volume 529*, page 208–215, Aachen, DEU. CEUR-WS. (Cited on p. 103)

[OECD, 2017] OECD (2017). *PISA 2015 assessment and analytical framework: Science, reading, mathematic, financial literacy and collaborative problem solving*. Organisation for Economic Co-operation and Development Publishing. (Cited on p. 89, 90)

[OED, 2024] OED (2024). *Oxford English Dictionary*. Oxford University Press. (Cited on p. 21, 88, 90, 92, 163, 165, 166, 169)

[Olivares-Alarcos et al., 2023a] Olivares-Alarcos, A., Andriella, A., Foix, S., and Alenyà, G. (2023a). Robot explanatory narratives of collaborative and adaptive experiences. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11964–11971. (Cited on p. 8, 130, 140)

[Olivares-Alarcos et al., 2019a] Olivares-Alarcos, A., Beßler, D., Khamis, A., Goncalves, P., Habib, M. K., Bermejo-Alonso, J., Barreto, M., Diab, M., Rosell, J., Quintas, J., Olszewska, J., Nakawala, H., Pignaton, E., Gyrard, A., Borgo, S., Alenyà, G., Beetz, M., and Li, H. (2019a). A review and comparison of ontology-based approaches to robot autonomy. *The Knowledge Engineering Review*, 34:e29. (Cited on p. 7, 8, 85, 112, 129)

[Olivares-Alarcos et al., 2024] Olivares-Alarcos, A., Canal, G., Foix, S., and Alenyà, G. (2024). Ontological modeling and reasoning for comparison and contrastive explanation of robot plans. In *Proceedings of the 23rd International Conference on Autonomous Agents and MultiAgent Systems*, page submitted. International Foundation for Autonomous Agents and Multiagent Systems. (Cited on p. 7, 8)

[Olivares-Alarcos et al., 2019b] Olivares-Alarcos, A., Foix, S., and Alenyà, G. (2019b). Force-based human intention inference (zenodo). (Cited on p. 44, 47)

[Olivares-Alarcos et al., 2019c] Olivares-Alarcos, A., Foix, S., and Alenyà, G. (2019c). On inferring intentions in shared tasks for industrial collaborative robots. *Electronics*, 8(11):1306. (Cited on p. 7, 97)

[Olivares-Alarcos et al., 2023b] Olivares-Alarcos, A., Foix, S., and Alenyà, G. (2023b). *Time-to-contact for robot safety stop in close collaborative tasks*, pages 87–104. Control, Robotics and Sensors. Institution of Engineering and Technology. (Cited on p. 7)

[Olivares-Alarcos et al., 2022] Olivares-Alarcos, A., Foix, S., Borgo, S., and Guillem Alenyà, (2022). Ocra – an ontology for collaborative robotics and adaptation. *Computers in Industry*, 138:103627. (Cited on p. 7, 112, 113, 130)

[Oliveira et al., 2007] Oliveira, F. F., Antunes, J. C., and Guizzardi, R. S. (2007). Towards a collaboration ontology. In *Proc. of the 2nd Brazilian Workshop on Ontologies and Metamodels for Software and Data Engineering*. João Pessoa. (Cited on p. 89, 90)

[Olszewska et al., 2017] Olszewska, J. I., Barreto, M., Bermejo-Alonso, J., Carbonera, J., Chibani, A., Fiorini, S., Goncalves, P., Habib, M., Khamis, A., Olivares-Alarcos, A., de Freitas,

E. P., Prestes, E., Ragavan, S. V., Redfield, S., Sanz, R., Spencer, B., and Li, H. (2017). Ontology for autonomous robotics. In *26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 189–194. (Cited on p. 112)

[Oltramari, 2019] Oltramari, A. (2019). Artificial intelligence within the bounds of ontological reason. In Borgo, S., Ferrario, R., and Masolo, C., editors, *Ontology Makes Sense*, pages 37–48. IOS Press. (Cited on p. 2, 3)

[Ortmann and Kuhn, 2010] Ortmann, J. and Kuhn, W. (2010). Affordances as qualities. In *Proceedings of the International Conference on Formal Ontology in Information Systems (FOIS)*, pages 117–130. IOS Press. (Cited on p. 162)

[Pan et al., 2019] Pan, M. K., Knoop, E., Bächer, M., and Niemeyer, G. (2019). Fast handovers with a robot character: Small sensorimotor delays improve perceived qualities. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6735–6741. (Cited on p. 97)

[Papadimitriou, 2003] Papadimitriou, C. H. (2003). Computational complexity. In *Encyclopedia of Computer Science*, page 260–265. John Wiley and Sons Ltd. (Cited on p. 19)

[Paulius and Sun, 2019] Paulius, D. and Sun, Y. (2019). A survey of knowledge representation in service robotics. *Robotics and Autonomous Systems*, 118:13–30. (Cited on p. 16)

[Pérez et al., 2006] Pérez, J., Arenas, M., and Gutierrez, C. (2006). Semantics and complexity of sparql. In *The Semantic Web - ISWC 2006*, pages 30–43, Berlin, Heidelberg. Springer Berlin Heidelberg. (Cited on p. 103)

[Persson et al., 2010] Persson, J., Gallois, A., Bjoerkelund, A., Hafdell, L., Haage, M., Malec, J., Nilsson, K., and Nugues, P. (2010). A knowledge integration framework for robotics. In *ISR 2010 (41st International Symposium on Robotics) and ROBOTIK 2010 (6th German Conference on Robotics)*, pages 1–8. (Cited on p. 173)

[Perzylo et al., 2019a] Perzylo, A., Grothoff, J., Lucio, L., Weser, M., Malakuti, S., Venet, P., Aravantinos, V., and Deppe, T. (2019a). Capability-based semantic interoperability of manufacturing resources: A basys 4.0 perspective. *IFAC-PapersOnLine*, 52(13):1590–1596. 9th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2019. (Cited on p. 164)

[Perzylo et al., 2019b] Perzylo, A., Rickert, M., Kahl, B., Somani, N., Lehmann, C., Kuss, A., Profanter, S., Beck, A. B., Haage, M., Rath Hansen, M., Nibe, M. T., Roa, M. A., Sörnmo, O., Gestegård Robertz, S., Thomas, U., Veiga, G., Topp, E. A., Kessler, I., and Danzer, M. (2019b). Smerobotics: Smart robots for flexible manufacturing. *IEEE Robotics & Automation Magazine*, 26(1):78–90. (Cited on p. 173)

[Peternel et al., 2018] Peternel, L., Tsagarakis, N., Caldwell, D., and Ajoudani, A. (2018). Robot adaptation to human physical fatigue in human–robot co-manipulation. *Autonomous Robots*, 42(5):1011–1021. (Cited on p. 43)

[Qu et al., 2018] Qu, C., Qi, W.-Y., and Wu, P. (2018). A high precision and efficient time-to-collision algorithm for collision warning based v2x applications. In *2018 2nd International Conference on Robotics and Automation Sciences (ICRAS)*, pages 1–5. (Cited on p. 66)

[Raiola et al., 2018] Raiola, G., Restrepo, S. S., Chevalier, P., Rodriguez-Ayerbe, P., Lamy, X., Tliba, S., and Stulp, F. (2018). Co-manipulation with a library of virtual guiding fixtures. *Autonomous Robots*, 42(5):1037–1051. (Cited on p. 43)

[Ramos et al., 2018a] Ramos, F., Scrob, C. O., Vázquez, A. S., Fernández, R., and Olivares-Alarcos, A. (2018a). Skill-oriented designer of conceptual robotic structures. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5679–5684. (Cited on p. 112)

[Ramos et al., 2018b] Ramos, F., Vázquez, A. S., Fernández, R., and Olivares-Alarcos, A. (2018b). Ontology based design, control and programming of modular robots. *Integrated Computer-Aided Engineering*, 25(2):173–192. (Cited on p. 112)

[Rayner et al., 2016] Rayner, K., Schotter, E. R., Masson, M. E. J., Potter, M. C., and Treiman, R. (2016). So much to read, so little time: How do we read, and can speed reading help? *Psychological Science in the Public Interest*, 17(1):4–34. (Cited on p. 148)

[Rector and Noy, 2006] Rector, A. and Noy, N. (2006). Defining n-ary relations on the semantic web. W3C note, W3C. https://www.w3.org/TR/2006/NOTE-swbp-n-aryRelations-20060412/. (Cited on p. 98)

[Riva and Riva, 2019] Riva, G. and Riva, E. (2019). Sarafun: Interactive robots meet manufacturing industry. *Cyberpsychology, Behavior, and Social Networking*, 22(4):295–296. (Cited on p. 173)

[Ros et al., 2010] Ros, R., Lemaignan, S., Sisbot, E. A., Alami, R., Steinwender, J., Hamann, K., and Warneken, F. (2010). Which one? grounding the referent based on efficient human-robot interaction. In *19th IEEE International Symposium in Robot and Human Interactive Communication (RO-MAN)*, pages 570–575. (Cited on p. 28, 29, 30, 33)

[Rosenfeld, 2021] Rosenfeld, A. (2021). Better metrics for evaluating explainable artificial intelligence. In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '21, page 45–50, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems. (Cited on p. 148)

[Rosenstrauch et al., 2018] Rosenstrauch, M. J., Pannen, T. J., and Krüger, J. (2018). Human robot collaboration - using kinect v2 for iso/ts 15066 speed and separation monitoring. *Procedia CIRP*, 76:183 – 186. 7th CIRP Conference on Assembly Technologies and Systems (CATS 2018). (Cited on p. 66)

[Rosenthal et al., 2016] Rosenthal, S., Selvaraj, S. P., and Veloso, M. M. (2016). Verbalization: Narration of autonomous robot experience. In *IJCAI*, volume 16, pages 862–868. (Cited on p. 112, 130)

[Roy and Edan, 2020] Roy, S. and Edan, Y. (2020). Investigating joint-action in short-cycle repetitive handover tasks: The role of giver versus receiver and its implications for human-robot collaborative system design. *International Journal of Social Robotics*, 12(5):973–988. (Cited on p. 40)

[Rozo et al., 2013] Rozo, L., Calinon, S., Caldwell, D., Jiménez, P., and Torras, C. (2013). Learning collaborative impedance-based robot behaviors. In *Proceedings of the AAAI conference on artificial intelligence*, volume 27. (Cited on p. 97)

[Rozo et al., 2016] Rozo, L., Calinon, S., Caldwell, D. G., Jiménez, P., and Torras, C. (2016). Learning physical collaborative robot behaviors from human demonstrations. *IEEE Transactions on Robotics*, 32(3):513–527. (Cited on p. 43)

[Rusu et al., 2009] Rusu, R. B., Marton, Z. C., Blodow, N., Holzbach, A., and Beetz, M. (2009). Model-based and learned semantic object labeling in 3d point cloud maps of kitchen environments. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3601–3608. (Cited on p. 162)

[Salustri, 2000] Salustri, F. A. (2000). Ontological commitments in knowledge-based design software: A progress report. In Finger, S., Tomiyama, T., and Mäntylä, M., editors, *Knowledge Intensive Computer Aided Design: IFIP TC5 WG5.2 Third Workshop on Knowledge Intensive CAD December 1–4, 1998, Tokyo, Japan*, pages 41–72. Springer US. (Cited on p. 163)

[Salvador and Chan, 2007] Salvador, S. and Chan, P. (2007). Toward accurate dynamic time warping in linear time and space. *Intelligent Data Analysis*, 11(5):561–580. (Cited on p. 48)

[Sampath Kumar et al., 2019] Sampath Kumar, V. R., Khamis, A., Fiorini, S., Carbonera, J. L., Olivares-Alarcos, A., Habib, M., Goncalves, P., Li, H., and Olszewska, J. I. (2019). Ontologies for industry 4.0. *The Knowledge Engineering Review*, 34:e17. (Cited on p. 86, 112)

[Saxena et al., 2015] Saxena, A., Jain, A., Sener, O., Jami, A., Misra, D. K., and Koppula, H. S. (2015). Robobrain: Large-scale knowledge engine for robots. arXiv preprint. (Cited on p. 170, 177)

[Schlenoff et al., 2012] Schlenoff, C., Prestes, E., Madhavan, R., Goncalves, P., Li, H., Balakirsky, S., Kramer, T., and Migueláñez, E. (2012). An ieee standard ontology for robotics and automation. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1337–1342. (Cited on p. 85, 86, 112, 129, 170, 171, 173, 174, 175)

[Scimmi et al., 2021] Scimmi, L. S., Melchiorre, M., Troise, M., Mauro, S., and Pastorelli, S. (2021). A practical and effective layout for a safe human-robot collaborative assembly task. *Applied Sciences*, 11(4). (Cited on p. 85)

[Seifert et al., 2018] Seifert, B., Korn, K., Hartmann, S., and Uhl, C. (2018). Dynamical component analysis (dyca): Dimensionality reduction for high-dimensional deterministic time-series. In *2018 IEEE 28th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6. (Cited on p. 49)

[Sgorbissa et al., 2018] Sgorbissa, A., Papadopoulos, I., Bruno, B., Koulouglioti, C., and Recchiuto, C. (2018). Encoding guidelines for a culturally competent robot for elderly care. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1988–1995. (Cited on p. 29, 32)

[Shokoohi-Yekta et al., 2017] Shokoohi-Yekta, M., Hu, B., Jin, H., Wang, J., and Keogh, E. (2017). Generalizing dtw to the multi-dimensional case requires an adaptive approach. *Data mining and knowledge discovery*, 31(1):1–31. (Cited on p. 48)

[Silverman, 1992] Silverman, B. G. (1992). Human-computer collaboration. *Human-Computer Interaction*, 7(2):165–196. (Cited on p. 89, 90)

[Sirin et al., 2007] Sirin, E., Parsia, B., Grau, B. C., Kalyanpur, A., and Katz, Y. (2007). Pellet: A practical owl-dl reasoner. *Journal of Web Semantics*, 5(2):51–53. (Cited on p. 176)

[Sisbot et al., 2011] Sisbot, E. A., Ros, R., and Alami, R. (2011). Situation assessment for human-robot interactive object manipulation. In *20th IEEE International Symposium in Robot and Human Interactive Communication (RO-MAN)*, pages 15–20. (Cited on p. 29, 30)

[Smit et al., 2000] Smit, B., Burton, I., Klein, R. J., and Wandel, J. (2000). An anatomy of adaptation to climate change and variability. *Climatic Change*, 45(1):223–251. (Cited on p. 92, 93)

[Smit and Wandel, 2006] Smit, B. and Wandel, J. (2006). Adaptation, adaptive capacity and vulnerability. *Global Environmental Change*, 16(3):282 – 292. (Cited on p. 92, 93)

[Smith et al., 2019] Smith, B., Ameri, F., Cheong, H., Kiritsis, D., Sormaz, D., Will, C., and Otte, J. N. (2019). A first-order logic formalization of the industrial ontologies foundry signature using basic formal ontology. In *Proceedings of the International Workshop on Formal Ontologies meet Industry (FOMI) at The Joint Ontology Workshops (JOWO)*. CEUR-WS. (Cited on p. 86)

[Someshwar and Edan, 2017] Someshwar, R. and Edan, Y. (2017). Givers & receivers perceive handover tasks differently: Implications for human-robot collaborative system design. (Cited on p. 40)

[Someshwar and Kerner, 2013] Someshwar, R. and Kerner, Y. (2013). Optimization of waiting time in h-r coordination. In *2013 IEEE International Conference on Systems, Man, and Cybernetics*, pages 1918–1923. (Cited on p. 40)

[Someshwar et al., 2012] Someshwar, R., Meyer, J., and Edan, Y. (2012). A timing control model for h-r synchronization. *IFAC Proceedings Volumes*, 45(22):698–703. (Cited on p. 40)

[Song et al., 2018] Song, W., Yang, Y., Fu, M., Qiu, F., and Wang, M. (2018). Real-time obstacles detection and status classification for collision warning in a vehicle active safety system. *IEEE Transactions on Intelligent Transportation Systems*, 19(3):758–773. (Cited on p. 66)

[Spyns et al., 2008] Spyns, P., Tang, Y., and Meersman, R. (2008). An ontology engineering methodology for dogma. *Applied Ontology*, 3(1-2):13–39. (Cited on p. 87, 130)

[Sridharan, 2023] Sridharan, M. (2023). Integrated knowledge-based reasoning and data-driven learning for explainable agency in robotics. In Aha, D. and Tulli, S., editors, *Explainable Agency in Artificial Intelligence: Research and Practice*, pages 43–70. CRC Press. (Cited on p. 6, 130, 157)

[Stenmark et al., 2017] Stenmark, M., Haage, M., Topp, E. A., and Malec, J. (2017). Supporting semantic capture during kinesthetic teaching of collaborative industrial robots. In *2017 IEEE International Conference on Semantic Computing (ICSC)*, pages 366–371. (Cited on p. 29, 33)

[Stenmark and Malec, 2013] Stenmark, M. and Malec, J. (2013). Knowledge-based industrial robotics. In *Proceedings of the Twelfth Scandinavian Conference on Artificial Intelligence*, pages 265–274. IOS Press. (Cited on p. 170, 173)

[Stenmark and Malec, 2015] Stenmark, M. and Malec, J. (2015). Knowledge-based instruction of manipulation tasks for industrial robotics. *Robotics and Computer-Integrated Manufacturing*, 33:56 – 67. (Cited on p. 86, 112)

[Stenmark et al., 2015] Stenmark, M., Malec, J., and Stolt, A. (2015). From high-level task descriptions to executable robot code. In Filev, D., Jabłkowski, J., Kacprzyk, J., Krawczak, M., Popchev, I., Rutkowski, L., Sgurev, V., Sotirova, E., Szynkarczyk, P., and Zadrozny, S., editors, *Intelligent Systems'2014*, pages 189–202. Springer International Publishing. (Cited on p. 29, 32)

[Stipancic et al., 2016] Stipancic, T., Jerbic, B., and Curkovic, P. (2016). A context-aware approach in realization of socially intelligent industrial robots. *Robotics and Computer-Integrated Manufacturing*, 37:79 – 89. (Cited on p. 86)

[Studer et al., 1998] Studer, R., Benjamins, V., and Fensel, D. (1998). Knowledge engineering: Principles and methods. *Data and Knowledge Engineering*, 25(1-2):161–197. (Cited on p. 2)

[Su et al., 2018] Su, B., Ding, X., Wang, H., and Wu, Y. (2018). Discriminative dimensionality reduction for multi-dimensional sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(1):77–91. (Cited on p. 49)

[Suh et al., 2007] Suh, I. H., Lim, G. H., Hwang, W., Suh, H., Choi, J.-H., and Park, Y.-T. (2007). Ontology-based multi-layered robot knowledge framework (omrkf) for robot intelligence. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 429–436. (Cited on p. 170, 177)

[Tenorth et al., 2014] Tenorth, M., Bartels, G., and Beetz, M. (2014). Knowledge-based specification of robot motions. In *Proceedings of the 21st European Conference on Artificial Intelligence (ECAI)*, page 873–878. IOS Press. (Cited on p. 29, 31, 33)

[Tenorth and Beetz, 2009] Tenorth, M. and Beetz, M. (2009). Knowrob — knowledge processing for autonomous personal robots. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4261–4266. (Cited on p. 86, 87, 137, 170, 172)

[Tenorth and Beetz, 2012] Tenorth, M. and Beetz, M. (2012). A unified representation for reasoning about robot actions, processes, and their effects on objects. In *2012*

IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 1351–1358. (Cited on p. 29, 31)

[Tenorth and Beetz, 2013] Tenorth, M. and Beetz, M. (2013). Knowrob: A knowledge processing infrastructure for cognition-enabled robots. *The International Journal of Robotics Research*, 32(5):566–590. (Cited on p. 175)

[Tenorth and Beetz, 2017] Tenorth, M. and Beetz, M. (2017). Representations for robot knowledge in the knowrob framework. *Artificial Intelligence*, 247:151–169. (Cited on p. 172)

[Tenorth et al., 2010a] Tenorth, M., Kunze, L., Jain, D., and Beetz, M. (2010a). Knowrob-map - knowledge-linked semantic object maps. In *2010 10th IEEE-RAS International Conference on Humanoid Robots*, pages 430–435. (Cited on p. 29, 32)

[Tenorth et al., 2010b] Tenorth, M., Nyga, D., and Beetz, M. (2010b). Understanding and executing instructions for everyday manipulation tasks from the world wide web. In *2010 IEEE International Conference on Robotics and Automation*, pages 1486–1491. (Cited on p. 29, 33)

[Terveen, 1995] Terveen, L. G. (1995). Overview of human-computer collaboration. *Knowledge-Based Systems*, 8(2-3):67–81. (Cited on p. 89, 90)

[Thosar et al., 2018] Thosar, M., Zug, S., Skaria, A. M., and Jain, A. (2018). A review of knowledge bases for service robots in household environments. In *6th International Workshop on Artificial Intelligence and Cognition*, pages 98–110. CEUR-WS. (Cited on p. 16, 170)

[Tiddi et al., 2017] Tiddi, I., Bastianelli, E., Bardaro, G., d'Aquin, M., and Motta, E. (2017). An ontology-based approach to improve the accessibility of ros-based robotic systems. In *Proceedings of the International Conference on Knowledge Capture*, pages 13:1–13:8. Association for Computing Machinery. (Cited on p. 164)

[Tiddi et al., 2015] Tiddi, I., d'Aquin, M., and Motta, E. (2015). An ontology design pattern to define explanations. In *Proceedings of the International Conference on Knowledge Capture*. Association for Computing Machinery. (Cited on p. 5)

[Tipping and Bishop, 1999] Tipping, M. E. and Bishop, C. M. (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 61(3):611–622. (Cited on p. 49)

[Topp et al., 2018] Topp, E. A., Stenmark, M., Ganslandt, A., Svensson, A., Haage, M., and Malec, J. (2018). Ontology-based knowledge representation for increased skill reusability in industrial robots. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5672–5678. (Cited on p. 29, 34, 173)

[Torres et al., 2015] Torres, P. M. B., Gonçalves, P. J. S., and Martins, J. M. M. (2015). Robotic motion compensation for bone movement, using ultrasound images. *Industrial Robot: An International Journal*, 42(5):466–474. (Cited on p. 32)

[Tsarouchi et al., 2017] Tsarouchi, P., Michalos, G., Makris, S., Athanasatos, T., Dimoulas, K., and Chryssolouris, G. (2017). On a human–robot workplace design and task allocation system. *International Journal of Computer Integrated Manufacturing*, 30(12):1272–1279. (Cited on p. 40)

[Tulving, 1972] Tulving, E. (1972). Episodic and semantic memory. In Tulving, E. and Donaldson, W., editors, *Organization of memory*, pages 381–403. Academic Press. (Cited on p. 5, 110, 112)

[Tulving, 2002] Tulving, E. (2002). Episodic memory: From mind to brain. *Annual Review of Psychology*, 53(1):1–25. (Cited on p. 5, 112)

[Turvey, 1992] Turvey, M. T. (1992). Affordances and prospective control: An outline of the ontology. *Ecological Psychology*, 4(3):173–187. (Cited on p. 162)

[Umbrico et al., 2020] Umbrico, A., Orlandini, A., and Cesta, A. (2020). An ontology for human-robot collaboration. *Procedia CIRP*, 93:1097 – 1102. 53rd CIRP Conference on Manufacturing Systems 2020. (Cited on p. 85, 86, 89, 90, 112)

[Uschold and Gruninger, 1996] Uschold, M. and Gruninger, M. (1996). Ontologies: principles, methods and applications. *The Knowledge Engineering Review*, 11(2):93–136. (Cited on p. 18)

[Vernon, 2014] Vernon, D. (2014). *Artificial cognitive systems: A primer*. MIT Press. (Cited on p. 20)

[Vernon et al., 2007] Vernon, D., Metta, G., and Sandini, G. (2007). A survey of artificial cognitive systems: Implications for the autonomous development of mental capabilities in computational agents. *IEEE transactions on evolutionary computation*, 11(2):151–180. (Cited on p. 20)

[Vicentini, 2020] Vicentini, F. (2020). Terminology in safety of collaborative robotics. *Robotics and Computer-Integrated Manufacturing*, 63:101921. (Cited on p. 84, 85, 89, 90, 95, 96)

[Vicentini et al., 2014] Vicentini, F., Giussani, M., and Tosatti, L. M. (2014). Trajectory-dependent safe distances in human-robot interaction. In *Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA)*, pages 1–4. (Cited on p. 64, 66)

[Villalobos et al., 2018] Villalobos, K., Diez, B., Illarramendi, A., Goñi, A., and Blanco, J. M. (2018). I4tsrs: A system to assist a data engineer in time-series dimensionality reduction in industry 4.0 scenarios. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, page 1915–1918. (Cited on p. 49)

[Villani et al., 2018] Villani, V., Pini, F., Leali, F., and Secchi, C. (2018). Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications. *Mechatronics*, 55:248–266. (Cited on p. 40)

[Villani et al., 2019] Villani, V., Sabattini, L., Loch, F., Vogel-Heuser, B., and Fantuzzi, C. (2019). A general methodology for adapting industrial hmis to human operators. *IEEE Transactions on Automation Science and Engineering*, pages 1–12. (Cited on p. 85)

[Vogel and Elkmann, 2017] Vogel, C. and Elkmann, N. (2017). Novel safety concept for safeguarding and supporting humans in human-robot shared workplaces with high-payload robots in industrial applications. In *Proceedings of the Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, page 315–316. Association for Computing Machinery. (Cited on p. 64, 65)

[Waibel et al., 2011] Waibel, M., Beetz, M., Civera, J., D'Andrea, R., Elfring, J., Gálvez-López, D., Häussermann, K., Janssen, R., Montiel, J., Perzylo, A., Schießle, B., Tenorth, M., Zweigle, O., and De Molengraft, R. V. (2011). Roboearth – a world wide web for robots. *IEEE Robotics & Automation Magazine*, 18(2):69–82. (Cited on p. 172)

[Wang et al., 2002] Wang, A. Y., Sable, J. H., and Spackman, K. A. (2002). The SNOMED clinical terms development process: refinement and analysis of content. pages 845–849. (Cited on p. 175)

[Wang et al., 2019] Wang, L., Gao, R., Váncza, J., Krüger, J., Wang, X., Makris, S., and Chryssolouris, G. (2019). Symbiotic human-robot collaborative assembly. *CIRP Annals*, 68(2):701–726. (Cited on p. 40)

[Warnier et al., 2012] Warnier, M., Guitton, J., Lemaignan, S., and Alami, R. (2012). When the robot puts itself in your shoes. managing and exploiting human and robot beliefs. In *21st IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 948–954. (Cited on p. 29, 32)

[Webb et al., 2023] Webb, N., Giuliani, M., and Lemaignan, S. (2023). Sogrin: a non-verbal dataset of social group-level interactions. In *32nd IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, pages 2632–2637. (Cited on p. 41)

[Yanco and Drury, 2004] Yanco, H. and Drury, J. (2004). Classifying human-robot interaction: an updated taxonomy. In *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)*, volume 3, pages 2841–2846. (Cited on p. 166)

[Yanco and Drury, 2002] Yanco, H. A. and Drury, J. L. (2002). A taxonomy for human-robot interaction. In *Proceedings of the AAAI Fall Symposium on Human-Robot Interaction*, pages 111–119. (Cited on p. 166)

[Yang et al., 2019] Yang, W., Zhang, X., Lei, Q., and Cheng, X. (2019). Research on longitudinal active collision avoidance of autonomous emergency braking pedestrian system (aeb-p). *Sensors*, 19(21):4671. (Cited on p. 66)

[Yazdani et al., 2018] Yazdani, F., Kazhoyan, G., Bozcuoğlu, A. K., Haidu, A., Bálint-Benczédi, F., Beßler, D., Pomarlan, M., and Beetz, M. (2018). Cognition-enabled framework for mixed human-robot rescue teams. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1421–1428. (Cited on p. 29, 33)

[Yu et al., 2016] Yu, K.-T., Bauza, M., Fazeli, N., and Rodriguez, A. (2016). More than a million ways to be pushed. a high-fidelity experimental dataset of planar pushing. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 30–37. (Cited on p. 41)

[Yuan et al., 2022] Yuan, L., Gao, X., Zheng, Z., Edmonds, M., Wu, Y. N., Rossano, F., Lu, H., Zhu, Y., and Zhu, S.-C. (2022). In situ bidirectional human-robot value alignment. *Science Robotics*, 7(68):eabm4183. (Cited on p. 1, 110, 128)

[Zanchettin et al., 2015] Zanchettin, A. M., Ceriani, N. M., Rocco, P., Ding, H., and Matthias, B. (2015). Safety in human-robot collaborative manufacturing environments: Metrics and control. *IEEE Transactions on Automation Science and Engineering*, 13(2):882–893. (Cited on p. 64, 66)

[Zanchettin et al., 2019] Zanchettin, A. M., Rocco, P., Chiappa, S., and Rossi, R. (2019). Towards an optimal avoidance strategy for collaborative robots. *Robotics and Computer-Integrated Manufacturing*, 59:47–55. (Cited on p. 85)

[Zhang et al., 2017] Zhang, H., Cheng, B., and Zhao, J. (2017). Extended tau theory for robot motion control. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5321–5326. (Cited on p. 66)

[Zhang and Zhu, 2018] Zhang, Q.-s. and Zhu, S.-c. (2018). Visual interpretability for deep learning: a survey. *Frontiers of Information Technology & Electronic Engineering*, 19(1):27–39. (Cited on p. 5)

[Zhao et al., 2018] Zhao, R., Drouot, A., and Ratchev, S. (2018). Classification of contact forces in human-robot collaborative manufacturing environments. *SAE International Journal of Materials and Manufacturing*, 11(05-11-01-0001):5–10. (Cited on p. 43)

”

..no soy nada,
nunca seré nada,
no puedo querer ser nada,
aparte de esto, tengo en mí todos los sueños del mundo..

..(y entre sueños me pregunto) ¿qué puedo saber de lo que seré, yo que no sé lo que soy? ¿ser lo que pienso?..¡pienso ser tantas cosas! ¡y hay tantos que piensan ser esas mismas cosas que no podemos ser tantos!

(aun así)..hoy estoy convencido como si supiese la verdad, lúcido como si estuviese por morir, y no tuviese más hermandad con las cosas que la de una despedida..

..¿genio? en este momento cien mil cerebros se creen en sueños genios como yo y la historia no recordará, ¿quién sabe?, ni uno..y sólo habrá un muladar para tantas futuras conquistas..

..no, no creo en mí..¡en tantos manicomios hay tantos locos con tantas certezas! yo, que no tengo ninguna..¿puedo estar en lo cierto?..no, en mí no creo..

..ni en mí ni en nada..derrame la naturaleza su sol y su lluvia sobre mi ardiente cabeza y que su viento me despeine..¿y después? que venga lo que viniere, o tiene que venir o no ha de venir..

— Fernando Pessoa
(Fragmentos de «*La Tabaquería*» y nexos añadidos)