DESCRIPCIO DELS COMPORTAMENTS DEL GENGHIS

Josep M Porta
Institut de Robòtica i Informática Industrial
UPC-CSIC

INDEX

Introducció 1

Nivell CONSTANTS 2

Nivell HOLD 3

- Llum-i-so 3
- Delta motors (0..5) 3
- Motors (0..5) 4
- A-delta-id (0..5) 6
- Inicialitza 6

Nivell BALANCE 7

- Alpha-balance 7
- Beta-balance 9

Nivell ADAPTATION 12

- Aixeca 12
- Aboca 12
- Pendent 14
- Combinacions de comportaments 15
  - Aixeca-Aboca 15
  - Aixeca-Antenes 16
  - Aboca-Antenes 16
  - Aixeca-Aboca-Antenes 17

Nivell COMPLIANCE 18

- No-vola (0..5) 18
- Sensor-a(0..5) 22
- Salta(0..5) 23
- Pas(0..5) 23

Nivell CLIFF 25

- Irs 25
- Explora-davant(0,1) 26
- Explora-darrera(4,5) 27
- Aplana-davant 27
- Aboca-davant 28
- Aplana-darrera 29
- Aboca-darrera 29

Nivell WALK 30

- Atura-a-balance(0..5) 30
- Regula-alçada(0..5) 30
- Torn(0..5) 30
- Primer-pas 33

Nivell DRIVE 34

- Canvi-pas 34
- Xoc 35
- Precipicis-davant 35
- Precipicis-darrera 37
- Obstacles-esq 39
- Obstacles-dreta 39

Annex I: Llistats 40

DESCRIPCIO DELS COMPORTAMENTS DEL GENGHIS

Introducció

Aquest document és el resum de tota la feina que hem estat duent a terme en el
amb l'objectiu d'aconseguir fer que el Genghis camini acceptablement sobre te
Durant tot aquest temps hem estat millorant els programes que controlen el Gen
formes de descompondre la tasca a assolir, millors estrarègies per a resoldre pr
ajustant centenars de paràmetres.

En aquests moments hem esgotat els recursos de memòria de que disposa el Gengh
un canvi de plataforma de treball. En aquest nou escenari el programa s'executarà
enviarà les ordres al robot i rebrà les dades dels sensors a través de línea sèrie.
implica el canvi de plataforma ja està encetat (comunicació entre el Genghis i un
però falta fer moltes coses:

- Definir la nova arquitectura que volem utilitzar (en principi similar a
  subsumpció).
- Definir el llenguatge en el que volem programar a partir d'ara: es poden fe
  Behavior Language.
- Programar el compilador d'aquest llenguatge: amb eines de depuració
  optimitzador de codi,...
- Transferir la feina feta fins ara al nou entorn de treball.

Abans de posar-nos en aquestes noves tasques (que ens poden portar molt te
documentar la feina feta amb un triple objectiu:

- Plasmar com s'implementen les nostres idees sobre com aconseguir un
  caminar adaptatiu per a un robot amb potes. Moltes d'aquestes idees ja les h
  articles que hem anat fent però aquí es pot veure la seva aplicació pràctica.
- Deixar constància de l'estat de la feina en aquests moments: quins object
  quines millores es podrien fer. També pretenem explicar quines alternatives
  què. Tot i això, durant aquest any hem fet moltes proves que no han que
  aquest treball.
- Documentar la nostra experiència de treball amb el Genghis. Per poder
  objectiu, hem hagut d'estudiar el comportament de quasi tots els sensor
  Genghis (només no hem utilitzat els sensors de calor). En diferents apartats
  es refexen els aspectes més interesants d'aquesta experiència que poden s
  futurs treballs amb el Genghis.

El text que s'acompanya té dos parts. La primera compren una explicació de tots
que actualment componen el nostre programa. Els expliquem agrupats per n
comportament, s'ha inclòs una llista de les seves entrades i sortides així com co
fan, els paràmetres que els caracteritzen i la seva relació amb altres comportame
d'aquest document no són més que els llistats originals dels diferents mòduls del p

Institut de Cibernètica
Barcelona, Novembre de 1995

Nivell CONSTANTS

En aquest nivell no hi ha res més que les declaracions de constants que s'us programa. Es tracta de:

- Nivell mínim de les bateries de motors.
- Estats dels autòmates dels sensors lògics de força dels motors alfa i beta.
- Límits físic i lògic (real i fictici) per als motors beta. Hi ha un a sèrie de re aquests dos paràmetres i el valor de standbeta (veure nivell BALANCE explicant al llarg d'aquest document però que, a mode de resum, detallem a

$$límitfísic <= 30$$
$$stbeta >= 0$$
$$límitlògic <= 54$$
$$stbeta < límitfísic <= límitlògic$$
$$límitlògic >= límitfísic + (límitfísic - stbeta)$$

- El nivell base de l'inclinòmetre (el valor que dona en plà)

També hi tenim la definició de macros que fan més clara i fàcil la programació. En

- Divisió (div a b c) c=a/b
- Multiplicació (mul a b c) c=a*b
- Autoincrement (++ a) a=a+1
- Autodecrement (-- a) a=a-1
- Generació de nombres aleatoris d'n bits (atzar n)
- Suma de sis números i divisió entera del resultat evitant overflows innecessaris.

  (suma-div n0 n1 n2 n3 n4 n5 d r t1 t2 t3)

  on:

  n(0..5) nombres a sumar. Han de ser dins l'interval [-30,54]
  d és el divisor
  r és on quedarà el resultat: r=(n0+n1+n2+n3+n4+n5) / d
  t1,t2,t3 són tres variables temporals

Nivell HOLD

Inclou, bàsicament, l'interfície amb els motors. S'unifica i facilita el fet poder ord
sigui per posició absoluta o relativa. També es regularitza l'accés a les lectures de
que té cada motor.

Llum-i-so

Mostra, a través dels LEDS, el nivell de les bateries. El rang en que es mou aq
de 42 (nivell que provoca la desconnexió per software del Genghis) fins a
poques vegades passa de 49). Aquest rang es mapeja en els LEDS de la següen

nivell<=43 -> 0000011
nivell =44 -> 0000101
nivell =45 -> 0001001
nivell =46 -> 0010001
nivell =47 -> 0100001
nivell>=48 -> 1000001

Aquest comportament també fa sonar el buzzer quan el nivell de bateries és m
estan massa calents, hi ha algunes potes fent massa força,...) En aquestes ci
aparar el Genghis i deixar-lo reposar.

Delta motors (0..5)

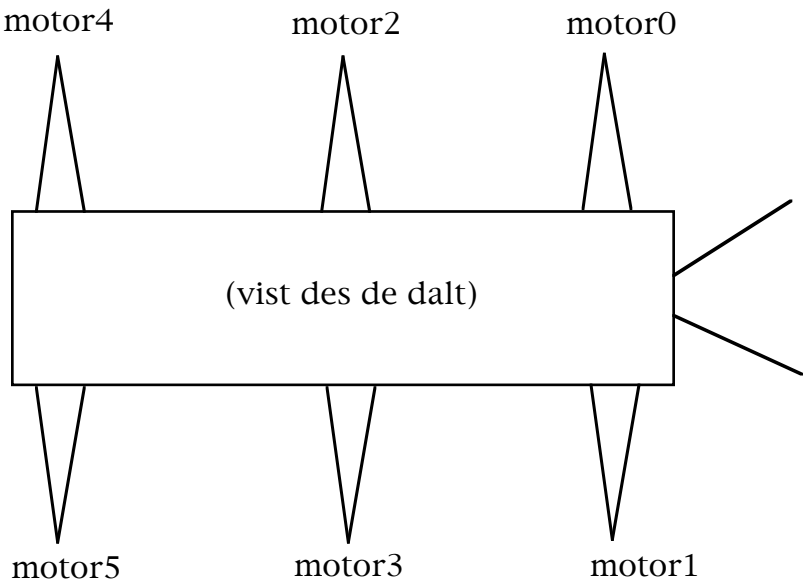
Entrades:

actual -> Darrera posició comandada al motor.
delta -> Variació de la posició del motor que es desitja fer.

Sortides

absolute-> Posició a la que ha d'anar a parar el motor: (actual+delta).

Tenim delta motors tant per als motors alfa com per als beta anomenat
adelta(0..5) i bdelta(0..5). La numeració dels motors es correspon a:



Motors (0..5)

Entrades:

absolute-> Noves ordres al motor (en posició absoluta).
delta-zero-> Canvis en el zero dels motors (en relatiu).

Sortides:

força-> Cada 0.02 segons s'indica el consum que fa el motor.
actual-> Cada 0.02 segons s'informa de la posició teòrica del motor (la processada).

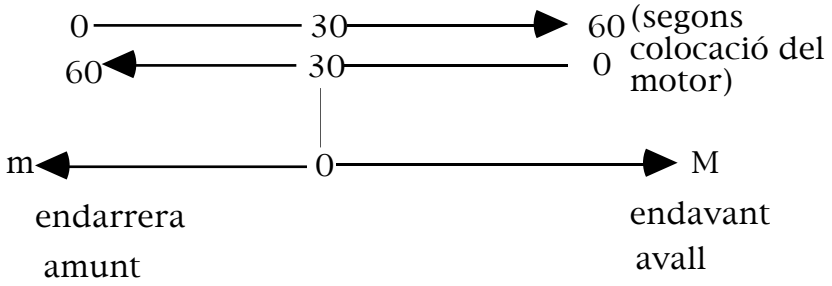
Tenim motors en beta (anomenats bmotors(0..5)) i motors en alfa (amotors(0 Els motor tenen dos classes de límits: els físics (o hard) i els lògics (o soft). haurien de ser 0 i 60 però defectes de mecànics fan que en algun cas s'hagi d interval. Aquests defectes són:

- Xoc amb el cos del robot: Passa en el moviment en alfa d'algunes potes eliminar aquests xocs amb el propi cos del robot per no confondre'ls a obstacles. Això implica calcular uns límits físics diferents per cada motor al del programa es poden veure quins valors cal posar per cada motor alfa.

- Cal evitar posicions extremes en que les potes, massa verticals, s'on incap força. Això només passa amb els motors beta als que prohibim posar-se en la part baixa del seu rang de moviments.

Per la seva banda, els límits lògics (o soft) constitueixen l'interfície d'accés als límits físics són ocults a l'usuari. Es defineixen com a l'interval [m,M] (amb (M-m)>=60 i (M-m) constant) i poden ser més amplis que els físics. A més, respecte la forma de referenciar les posicions dels motors per tal que el ze recorregut del motor (en principi al mig) i que positius/negatius signifiquin (independentment de com estigui col.locat el motor). Gràficament:

LIMITS FISICS:



LIMITS LOGICS

Per als motors alfa usem uns límits lògics de:

m=-30 (endarrera) M= 30 (endavant)

i per als motors beta:

m=-30 (amunt) M=45 (avall)

La funció dels límits lògics és la de superar, en certa mesura, les restriccions límits físics. Així, en el cas dels motors alfa, posicions prohibides pels lín assolides gracies a l'ajut de les altres potes i de l'alpha-balance. Per als motor cosa similar: Una pota enviada a la posicio M (45) no passa de la posició marc (25 en el nostre cas) però informa a la resta de comportaments de que ha a provoca una acció dels beta-balanços més important que si hagués declarat la Com més gran sigui la diferència

més exagerat serà aquest efecte tal com es veu en la següent taula:
(Vàlida només per potes frontals i posteriors)

Diferència entre límits	Acomodació extra
0	0
1	0
2	0
3	0
4	1
5	1
6	3
7	3
8	4
9	4
10	4
11	4
12	7
……	…..
20	11

L'acomodació extra es calcula acumulant els efectes dels diferents beta bala
sobre la pota en qüestió:

Lateral:

$$\left\lfloor \frac{(límitlògic - límitfísic)}{3 \cdot 2} \right\rfloor$$

Frontal:

$$\left\lfloor \frac{(límitlògic - límitfísic)}{2 \cdot 2} \right\rfloor$$

Global:

$$\left\lfloor \frac{(límitlògic - límitfísic)}{6} \right\rfloor$$

Els valors triats en el nostre cas són:

Límit físic -> 25
Límit lògic -> 45
Diferència -> 20
Acomodació extra -> 11
Posició final assolida 36 (= límit físic+acomodació extra=25+11)

Una posició final tan baixa ens garantitza adaptar-se força bé als forats del ter

Es possible canviar el zero (posició de referència) dels motors. Inicialment
meitat del recorregut teòric dels motors excepte en alguns motors beta. En a
zero de tal forma que, en la situació de repós (els motors alfa i beta en posi

potes estiguin en un mateix pla (defectes de construcció fan que aquesta n
necessaria). Canvis en els zeros fan que també es modifiquin els límits lògics (
la posició actual (sempre es dona en referència al zero en curs). Cal anar am
aquesta opció ja que valors exagerats de la posició del zero poden resultar e
algunes operacions dels balanços. Aquest inconvenient es podria soluciona
donat que en l'actualitat cap comportament varia els zeros dels motors no val
l'escasa memòria de que disposem en solventar-lo.

A-delta-id (0..5)

Entrada:

delta-> Variació que es vol fer a la posició del motor alfa corresponent.

Sortida:

delta-out-> El mateix que ha entrat.

No fa més que posar l'entrada a la sortida. Ens proporciona un nivell més
motors imprescindible alhora de montar una jerarquia no ambigüa de priorita
motors.

Inicialitza

Paràmetres d'usuari:

a (0..5)-> Posició que l'usuari desitja per als motors alfa 0..5.

b (0..5)-> El mateix per als motors beta 0..5.

Sortides

pa (0..5)-> Modificacions enviades als motors alfa 0..5.

pb (0..5)-> Modificacions enviades als motors beta 0..5

Es el procediment de prova de nivell HOLD. L'usuari pot especificar en quina
els motors a través dels paràmetres. Les ordres donades s'envien als motors.
les posicions dels motors aquest comportament no s'entera (no té entrades
interfereix. Si no es modifiquen els paràmetres a0,..a6, b0,..,b6 aquest compo
pensant-se que les potes estan tota l'estona en la posició 0 (posició inicial pe
beta).

Nivell BALANCE

Control dels 5 graus de llibertat en els que es pot moure el robot. S'intenta man equilibri per cadascun d'ells. Per caminar no caldrà mès que trencar periòdicamer deixar que els comportaments d'aquest nivell ho intentin arreglar.

Alpha-balance

Entrades:

apos(0..5)-> Posicions actuals dels alfa-motors.

alfa-zero-> Punt d'equilibri del alfa-balance global.

ab-lateral-> Punt d'equilibri de l'alfa-balance lateral. Negatiu orienta el robo i positiu a la dreta.

Sortides

s-e-> Modificacions a les posicions dels motors alfa de la banda esquerra.

s-d-> Modificacions a les posicions dels motors alfa de la banda dreta.

Aquest comportament inclou dos balanços:

- Alpha-balance global: intenta mantenir la mitjana de les posicions dels m alfa-zero. Calcula la mitjana de la posició actual dels motors alfa i, si no és ordena a tots el motors alfa moure's un valor que, si tots executesin desequilibri:

$$correcció_global = alfa_zero - \frac{\sum_{i=0}^{5} apos_i}{6}$$

- Alpha-balance lateral: intenta mantenir la diferència de la mitjana de l motors alfa de la banda esquerra i la dels motors alfa de la banda dreta igu cas de desequilibri divideix l'error entre dos i n'assigna una meitat (esquerra/dreta) del robot amb signes contraris per tal de cancel.lar el prob

$$correcció_esquerra = \frac{\left(\frac{(apos1 + apos3 + apos5)}{3} - \frac{(apos0 + apos2 + apos4)}{3}\right) - ab_lateral}{2}$$

$$correcció_dreta = -correcció_esquerra$$

Una bona forma de veure la combinació dels dos alfa-balanços és conside vectorial. Agrupant les potes per bandes (esquerra i dreta) passem d'un e dimensió sis (sis graus de llibertat en alfa) a un de dimensió dos. A cada pc correspon un punt en aquest pla que és: [*mitjana_posicions_esquerra, mitjana_posicion*]. El que fan els alfa balancos es desplaçar aquest punt fins al punt d'equil *ab_lateral) , (alfa_zero - ab_lateral)*]:

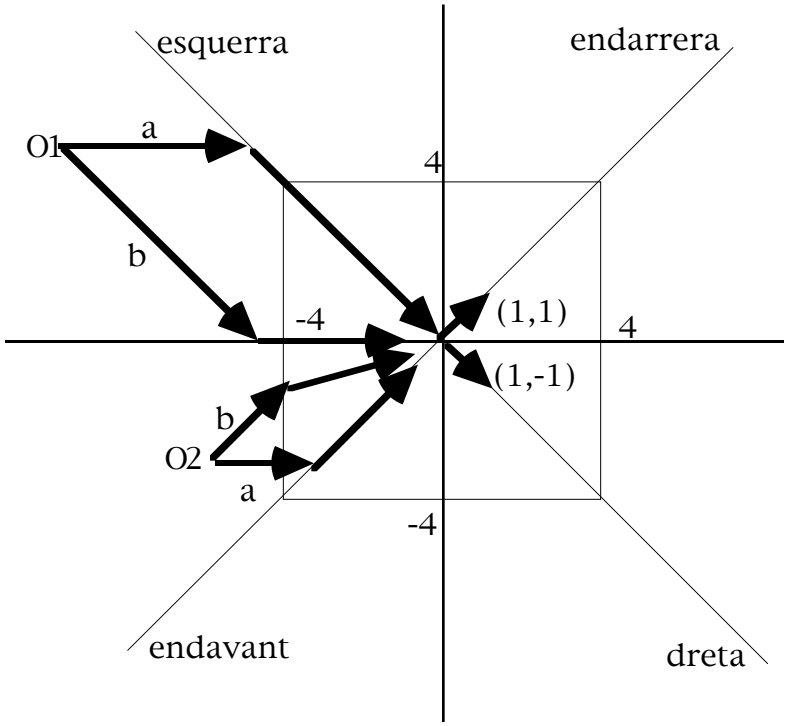
$$correcció = alfa_balance_global + alfa_balance_lateral =$$

$$= correcció_global \begin{pmatrix} \text{efecte en banda esquerra} \\ \text{efecte en banda dreta} \end{pmatrix} + correcció_esquerra \begin{pmatrix} \text{efecte en banda esq} \\ \text{efecte en banda dr} \end{pmatrix}$$

$$correcció = correcció_global \begin{pmatrix} 1 \\ 1 \end{pmatrix} + correcció_esquerra \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

Com es pot veure l'alpha-balance global i el lateral són components ortogonals
executar.
Abans de realitzar la suma es limita el rang de cadascuna de les components
cosa que restringeix el màxim moviment produït per l'alfa balance a [-4,4]. Aix
- L'alfa balance ha d'actuar d'una forma suau.
- Els moviments d'alfa balance han de ser petits (no més grans que la consta
inicialment posada a valor 10) per tal de no confondre altres comportament
moviment d'alpha-balance podria, per exemple, disparar el rebot (cosa no d

Aquesta retallada també es podria fer un cop feta la suma de les correccions
Això origina diferents trajectories fins arribar a l'equilibri tal com es veu en la s



O1,O2 dos situacions inicials
(0,0) equilibri si alfa_zero=0 i ab_lateral=0
(1,1) (-1,1) components de l'alfa-balance
Trajectòries a: restringint cada component per separat
Trajectòries b: restringint la suma

Ens hem quedat amb les trajectòries tipus 'a' ja que permeten, al principi del c
als dos balanços enlloc de només al més desequilibrat (com fan les trajectòries

Si el marge de tolerància de l'alpha-balance lateral és 0 llavors el robot camina
que hi ha estats dins del cicle de qualsevol patró de passes que no són equilibr
balance. Així, per exemple, si utilitzem un "tripot gait" podem, tot i caminar
desequilibris laterals dins l'interval:

$$\left[-\frac{2\ \$pas}{3}, \frac{2\ \$pas}{3}\right] \cong [-0.66\$pas,\ 0.66\$pa\$$$

on \$pas és una constant que indica el desplaçament sobre el valor de alfa-zer
quan una pota fa un pas. Si sempre caminem amb tripot gait, l'alpha-balance l
disparar-se per diferències dins aquest interval.
Per la seva banda, usant un "slow gait" l'interval de desequilibris a permetre se

$$\left[-\frac{6\ \$pas}{5},\ \ \frac{6\ \$pas}{5}\right] = [-1.2\$pas,\ \ 1.2\ \$pa\$$$

I el rang màxim de la diferència lateral és:

$$[-2\$pas\ \ 2\$pa\$$$

El marge que té assignat l'alpha-balance lateral en el programa és de:

$$[-\$pas\ \ \$pa\$$$

que sembla prou raonable ja que està en un valor intermedi respecte la m
possible i respecte les diferècies que s'haurien de permetre en tripot i en sl
petit sigui aquest inteval més fàcil serà girar (s'hauran de fer menys pas
determinat angle) però més difícil serà caminar recte.

Beta-balance

Paràmetres:

Standbeta-> Alçada mitjana del robot fixada per l'usuari.
bl-> Equilibri lateral fixat per l'usuari.
bf-> Equilibri frontal fixat per l'usuari.

Entrades:

b(0..5)-> Posicions dels motors beta.
bb-alçada-> Modifiació sobre l'alçada mitjana fixada per l'usuari.
bb-frontal-> Modificació sobre l'equilibri frontal fixat per l'usuari.
bb-lateral-> Modificació sobre l'equilibri lateral fixat per l'usuari.
a(0..1)-> Alteracions als càlculs del beta balance produïdes a causa del cont
amb algun obstacle. Es comenten més detalladament en comportament 'a
ADAPTATION.

Sortides:

s(0..5)-> Correccions a les posicions de cada pota

Aquest comportament inclou el control de tres equilibris a través dels motors

- Global: Intenta mantenir la mitjana de les posicions dels motors beta igual
alçada). Si hi ha alguna diferència respecte la mitjana desitjada es corret
moviment simultani de tots els motors beta de:

$$correcció_global = \left(mitjana_desitjada - mitjana_real\right) = \left[\left(stdbeta + bb_alçada\right) - \frac{\sum_{i=0}^{5} b_i}{6}\right]$$

- Lateral: Intenta mantenir la diferència entre la posició mitjana dels motc
esquerra i els del dret igual a (bl + bb-lateral). Si aquest valor és negatiu el
l'esquerra i si és positiu ho fa de la dreta. Si hi ha un error respecte la mi
corretgeix ordenant moviments de:

$$correcció_esquerra = \frac{\left(\left(bl + bb_lateral\right) - diferència_lateral\right)}{2}$$

$$correcció_dreta = -correcció_esquerra$$

on:

$$diferència_lateral = \frac{(b0 + b2 + b4)}{3} - \frac{(b1 + b3 + b5)}{3}$$

- Frontal: Intenta mantenir la diferència entre la posició mitjana de les potes posteriors igual a (bf + bb-frontal). Si aquest valor és negatiu el robot s'aixec es positiu ho fa del davant. En cas de desequilibri la forma de corretgir-lo és

$$correció_davant = \frac{\left((bf + bb_frontal) - diferència_frontal\right)}{2}$$

$$correcció_darrera = -correcció_davant$$

on:

$$diferència_frontal = \frac{(b0 + b1)}{2} - \frac{(b4 + b5)}{2}$$

Cadascun d'aquest processos té un marge de tolerància respecte el desequilibr a valor 3 per a tots). Quan aquest marge se sobrepassa s'anul.la aquesta tole valor 0) fins que es reestableix l'equilibri. D'aquesta forma s'evita dispara continuament i quan es fa s'assegura restablir l'equilibri completament (anar a l'interval de tolerància cosa que evita conflictes amb el comportament no COMPLIANCE).

Els tres beta balanços es poden veure (a l'igual que els alfa balanços) en t considerant que el comportament beta balance té sis sortides que van a cad beta (0..5):

$$correcció_bb = correcció_fontal + correcció_lateral + correcció_global =$$

$$= correcció_davant \cdot \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ -1 \\ -1 \end{pmatrix} + correcció_esquerra \cdot \begin{pmatrix} 1 \\ -1 \\ 1 \\ -1 \\ 1 \\ -1 \end{pmatrix} + correcció_global \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} =$$

$$= \begin{pmatrix} \text{correcció per beta } 0 \\ \text{correcció per beta } 1 \\ \text{correcció per beta } 2 \\ \text{correcció per beta } 3 \\ \text{correcció per beta } 4 \\ \text{correcció per beta } 5 \end{pmatrix}$$

Com es pot veure les tres components (frontal, lateral i global) són ortogona una d'elles no afectes a les altres). Es evident que ens falten tres vectors ort arribar a dimensió sis (tenim sis sortides i per tant sis graus de llibertat). Aque podrien ser:

$$\begin{pmatrix} 1 \\ -1 \\ 0 \\ 0 \\ -1 \\ 1 \end{pmatrix} \qquad \begin{pmatrix} 1 \\ -1 \\ -2 \\ 2 \\ 1 \\ -1 \end{pmatrix} \qquad \begin{pmatrix} 1 \\ 1 \\ -2 \\ -2 \\ 1 \\ 1 \end{pmatrix}$$

Cadascun d'ells defineix un nou equilibri que no controlem per que no tene especial. Si no fos per l'existència de topes no seria possible, per la simple balance, estar desequilibrats respecte aquest nous equilibris (són ortogonals a

del beta balance). Aquests tres graus de llibertat que no controlem són exac
falta per adaptar-se a una superfície en concret: tres potes ens determinen la
de llibertat controlats) i les altres tres han de poder moure's lliurement per a
(els tres no controlats).
Abans d'efectuar l'output definitiu de les correccions del beta balance cal li
enviar amb l'objectiu de que els balanços actuïn suaument.
Si aquest retall es fa sobre el resultat d'acumular els tres beta balanços corre
de l'espai vectorial que defineixen. Per exemple poden tenir un cas com:

$$3\cdot\begin{pmatrix}1\\1\\0\\0\\-1\\-1\end{pmatrix}+0\cdot\begin{pmatrix}1\\-1\\1\\-1\\1\\-1\end{pmatrix}+3\cdot\begin{pmatrix}1\\1\\1\\1\\1\\1\end{pmatrix}=\begin{pmatrix}6\\6\\3\\3\\0\\0\end{pmatrix}$$

I truncant totes les correccions dins l'interval [-4,4] queda:

$$(4\quad 4\quad 3\quad 3\quad 0\quad 0)$$

que no és linealment independent del tercer dels equilibris no contemplats:

$$(4\quad 4\quad 3\quad 3\quad 0\quad 0)\cdot\begin{pmatrix}1\\1\\-2\\-2\\1\\1\end{pmatrix}=-4\neq 0$$

I per tant és linealment independent dels tres equilibris controlats.
Així doncs, cal retallar cadascuna de les components (global, frontal i lat
(donada la simetria dels vectors que les representen el fet de retallar-les man
linealment independent de les altres).

El procés complet és:

- Calcular la correcció frontal i limitar-la a l'interval [-1,1].
- Calcular la correcció lateral i limitar-la a l'interval [-1,1].
- Calcular la correcció global i limitar-la a l'interval [-2,2] (atorguem més perque és el que més ajuda a compensar el comportament 'no-vo COMPLIANCE).
- Enviar a cada motor (per una sortida diferent) l'acumulació de l calculades.

Uns dels aspectes més crítics per al bon funcionament del programa és deter correcta a la que ha d'actuar el beta balance. Empíricament em vist q d'acomodació es prou bo un valor al voltant de 0.1 segons, en canvi, per c valors propers a 0.25 segons. Per aquesta raó llegim el dip 1 del dips de esquerra del robot) que indica si el robot camina o no per saber quin dels dos

Nivell ADAPTATION

Ajusta els punts d'equilibri dels balanços en funció de diversos sensors (antene inferior,...) per tal d'adaptar-se al terreny.

Aixeca

Entrades:

Zero-> Aportacions d'altres comportaments a la variació de l'alçada del robot

Sortides:

Canvi-stb-> Variació que volem sobre el standbeta fixat per l'usuari.

La idea en la que es basa aquest comportament és la d'aixecar el cos del robo determinat xoc amb el sensor de tacte inferior. En concret, es reacciona si el co en la zona central de la panxa o simultaneament en les dos zones extremes.
Mentre es detecta contacte s'incrementa la variació que s'envia al beta bala incrementar l'alçada mitjana desitjada ja que el beta balance s'encarregarà pràctica. Si es deixa de detectar el contacte durant més de mig segon, progressivament la variació abans esmentada fins a fer-la nul.la. Aquest procé més lent que el d'increment (va a la meitat de velocitat) per evitar recaure mol contacte ja detectat.
El que s'envia cap al beta balance com a modificació definitiva al registre star entre l'increment calculat per aquest comportament i el rebut a través de Inicialment aquest registre val 0 (no té cap influència) però algú el pot modific (indirectament) a la alçada mitjana del robot.
Aquest idea d'accés indirecte es pot repetir tants cops com desitgem aconsegu comportaments que van acumulant, de forma progressiva, les seves influènc paràmetre del sistema. A més tenim l'avantatge de que és un montatge molt senzill desactivar algun del mòduls de la cadena sense quasi repecutir en e estructura que utilitzem diversos cops en el programa.

Abans d'enviar la variació definitiva al beta balance la limitem a un interval que

(-stbeta) -> com a molt poc tindrem una mitjana d'alçada de 0 (stbeta+(-s representa uns pocs centímetres de separació entre el robot i el terra.

(límitfísic - stbeta) -> L'alçada màxima del robot vindrà marcada pel límit físic beta (stbeta + (límitfísic - stbeta) = límitfísic).

Aboca

Entrades:

Zero-> Aportacions d'altres comportaments per tal d'establir un nou equilibr

Sortides:

Canvi-inclinació-> Variació sobre l'equilibri del balance frontal. Subsumeix frontal del beta balance.

Si es detecta xoc amb un dels sensors de tacte dels extrems s'aixeca el ro corresponent. Si el xoc es simultani en els dos costats el resultat net és variaci hi hagués contacte).

La variació final que s'envia al beta balance és la suma entre del contingut del l'accés indirecte a bb-frontal) i la variació que ha calculat aquest comportame resultat a l'interval $[-2\cdot(límitfísic - stbeta), 2\cdot(límitfísic - stbeta)]$. Aquest interval assegu que cap pota sobrepassa el límit físic per causa de correccions del balance implicaria posar algunes potes sense contacte amb el terra).

Antenes

Sortides:

a0,a1-> Influència d'aquest comportament en el beta balance

Aquest comportament modifica alguns dels càlculs de les mitjanes efectuats amb l'objectiu de fer aixecar el robot del costat de l'antena que detecta co d'aquest comportament radica en que si, apujant el robot, deixem de sentir girar (nivell DRIVE) a causa d'obstacles poc alts.
En concret els balanços alterats són el frontal i el lateral i l'objectiu a assoli propera a l'antena que detecta el contacte al límit físic (el més estirada possib altres es col.loquin adequadament per efecte dels balanços. Les senyals a0 i a els beta-balanços frontal i lateral) als mateixos llocs en que apareixen les pos beta 0 i beta 1 respectivament. Els valors negatius de a0 i a1 fan que el beta-b aquestes potes estan massa aixecades i per tant les fa baixar (que és el que v mínims de les sortides a0 i a1 ha de ser tal que l'acció combinada del balance fasi passar del límit físic a les respectives potes (0 i 1). Aquest mínim es pot una situació equilibrada en la que totes les potes estan en standbeta i veient l'influència d'aquest comportament per tal de que una de les potes del dava físic:

$$posicio_pota_davant = stbeta + correcció_frontal + correcció_lateral = límitfísic$$

on correcció_frontal i correcció_lateral són les produïdes per l'efecte de le deduïm:

$$stbeta - \frac{m}{4} - \frac{m}{6} = límitfísic$$

on m és la màxima influència de les antenes que ha de ser:

$$m = -\frac{(límitfísic - stbeta)\cdot 12}{5}$$

Que es pot aproximar per defecte per:

$$m = -2\cdot(límitfísic - stbeta)$$

Quan actuen combinadament les dos antenes l'efecte del balance lateral es balance frontal es magnifica:

$$stbeta + correcció_frontal + correcció_lateral = stbeta - \frac{m}{2} + 0$$

Si 'm' fos el valor exacte que hem calculat les potes del davant depassarien stbeta<límitfísic):

$$stbeta + correcció_frontal + correcció_lateral = stbeta - \frac{m}{2} + 0 =$$

$$= stbeta + \frac{(límitfísic - stbeta) \cdot 12}{10} > límitfísic$$

però amb l'aproximació per defecte que hem agafat es queden just al límit:

$$stbeta + correccio_frontal + correccio_lateral = stbeta - \frac{m}{2} + 0 =$$

$$= stbeta + \frac{(límitfísic - stbeta) \cdot 2}{2} = límitfísic$$

Pendent

Sortides:

Nou-zero-> Canvi a fer en l'equilibri de l'alfa-balance global per tal que, pendent, s'intenti posar el centre de gravetat el més centrat possible di suport. Per altra banda, hi ha molts llocs del programa que utilitzen aques indicador fiable de l'inclinació actual del robot.

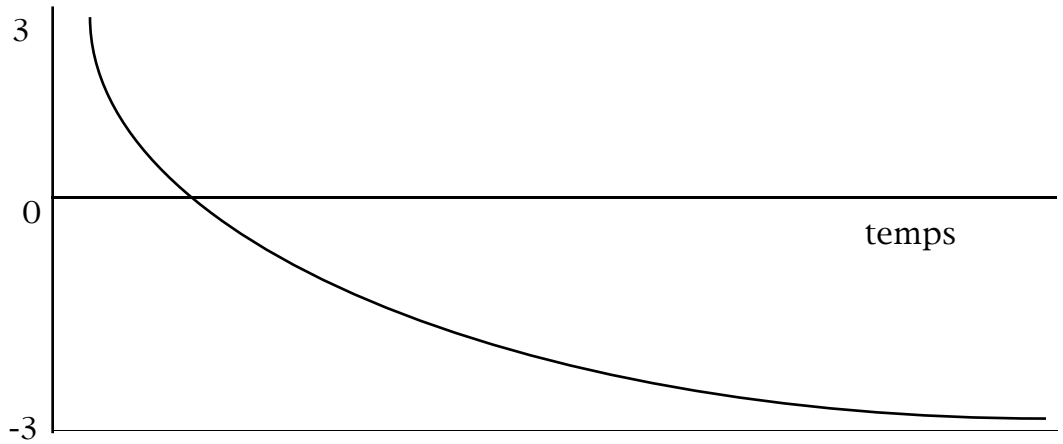
Aquest comportament intenta mantenir el centre de gravetat proper al cent suport definit per les potes (independentment del pendent en el que estem) una alta estabilitat.

La part interesant d'aquest comportament està en que utilitza l'inclinòmetre servit d'excusa per fer un estudi del funcionamet d'aquest sensor.

La primera cosa a destacar és que les seves lectures tenen un rang de 200 v voltant del 100 ([0..200]). Això fa que se superi amb escreix el valor 127 co problemes en el Behavior Language. En aquest tot nombre superior a 127 es c tot provocant una discontinuitat en les lectures del inclinòmetre.

Per evitar aquest problema ens hem definit una macro (llegir-inclinacio) q discontinuitat i ens dona lectures centrades en el zero (el rang de lectures pas on negatiu vol dir aixecat del davant i positiu aixecat del darrera).

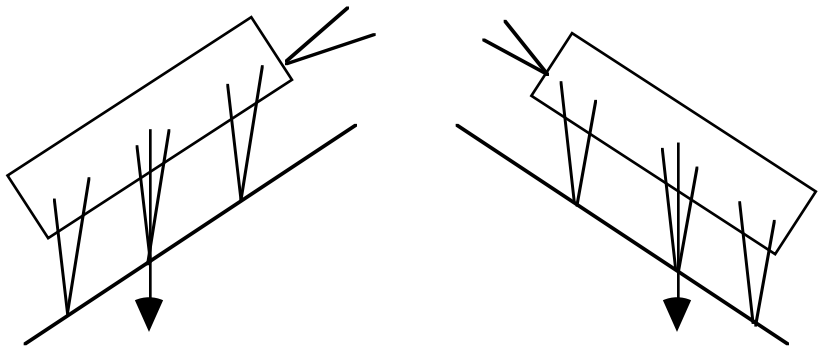
El següent problema que es detecta és que el zero de l'inclinòmetre (sortida robot està pla) varia en el temps (sembla tenir una relació amb l'escalfament una corba similar a (obviant el soroll):



Això és impossible de detectar per programa ja que un cop engegat el robot no estat fa poca estona (i per tant si està escalfat o no) ni tampoc sabem si estem engegar el Genghis en un pendent). La solució adoptada ha està la de prendre

referència el 0 tot i que al principi és un valor baix i quan fa una estona que e
un valor alt.

La sortida d'aquest comportament es calcula de la següent forma: si es fan mé
inclinòmetre concentrades al voltant un determinat valor llavors es divideix
(limitem el rang a [-33,33]) i es fa l'output corresponent a través del registre
registre s'usa com una indicació de l'estat d'inclinació actual del robot des de
programa però principalment s'usa per subsumir el registre alfa-zero del cor
balance per tal de guanyar estabilitat en les pendents (en una baixada les pote
en una pujada endarrera). Gràficament:



El fet d'esperar 9 lectures similars i de dividir per 3 els valors llegits fa que e
típic dels sensors reals i que pics bruscos i curts de l'inclinació no siguin tingu

Combinacions de comportaments

En aquest nivell (ADAPTATION) hem estat fent càlculs sobre màxim que diversos
podien influïr en els equilibris del robot. Però cal plantejar-se que passa si m
comportaments actua a la vegada (és possible que els màxims es rebassin).
Els problemes només poden apareixer entre comportaments que actuen sobre el
Aquests són:

Aixeca - Aboca: Conflicte en les potes anteriors o posteriors
Aixeca - Antenes: Conflicte en les potes anteriors
Aboca - Antenes: Conflicte en les potes anteriors
Aixeca - Aboca -Antenes: Conflicte en les potes anteriors

Passem a estudiar cada cas.

- Aixeca-Aboca:
  Aixeca posa les potes al *límit físic* i inclina obre un extrem i en tanca l'altre en *(lí*
  Per tant les potes més tancades estaran a:

  (limitfisic)+(límitfísic - stbeta)

  Si el límit lògic és menor que això i haurà potes que arribaran al tope cosa q
  es pugui assolir l'equilibri (ni el frontal ni el global). En el millor dels cas
  l'extrem de robot que no estan en el tope quedaran en un punt intermedi ent
  pel balance frontal i l'equilibri desitjat pel balance global. En aquest pun
  ordenats pels dos balanços es compensen (és una espècie de punt fix). Però

que les potes quedin oscil.lant entre dos posicions al voltant d'un inabast exemple, en una situació com:

$$posicions_potes = \begin{pmatrix} límitlògic - 2 \\ límitlògic - 2 \\ límitlògic \\ límitlògic \\ límitlògic \\ límitlògic \end{pmatrix} \qquad equilibris_desitjats = \begin{pmatrix} frontal = -5 \\ lateral = 0 \\ global = límitlògic \end{pmatrix}$$

tindrem una seqüència cíclica de moviments tal com:

$$\begin{pmatrix} límitlògic - 2 \\ límitlògic - 2 \\ límitlògic \\ límitlògic \\ límitlògic \\ límitlògic \end{pmatrix} \xrightarrow{\text{actua frontal} \begin{pmatrix} -1 \\ -1 \\ 0 \\ 0 \\ 1 \\ 1 \end{pmatrix}} \begin{pmatrix} límitlògic - 3 \\ límitlògic - 3 \\ límitlògic \\ límitlògic \\ límitlògic \\ límitlògic \end{pmatrix} \xrightarrow{\text{actua global} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}} \begin{pmatrix} límitlògic - 2 \\ límitlògic - 2 \\ límitlògic \\ límitlògic \\ límitlògic \\ límitlògic \end{pmatrix}$$

A més el fet de que unes potes arribin a un tope trenca la ortogonalitat entre aquesta està basada en el fet de que tots el moviments ordenats pel balan pràctica. En l'exemple anterior es pot veure que una actuació del beta-balance l'equilibri global (cosa que, en teoria, no hauria de passar).
Sembla raonable, doncs, definir un límit lògic superior a *(limitfisic)+(límitfísic - stb* d'estalviar-nos tots aquests problemes.

- Aixeca-Antenes

Quan el contacte es produeix amb les dos antenes és un cas igual al que acabe contacte es produeix només amb una antena llavors les posicions de les p menors a les que es produeixen en el cas anterior i per tant no dona probleme

- Aboca-Antenes

El conflicte es dona en les potes del davant quan hi ha contacte amb les anten de tacte frontal: tots dos comportamentes pretenen tancar les potes del dav darrera.
La posició màxima a la que es voldrà enviar les potes del davant serà:

efecte_antenes+efecte_aboca=
=límitfísic + (limitfísic - stbeta)

I per tant quedem dins el límit lògic triat en l'apartat Aixeca-Aboca.

- Aixeca-Aboca-Antenes

Es l'únic cas en que se solapen tres comportaments intentant tancar unes pot
La situació correspon a:

Sensor tacte del mig activat
Sensor tacte frontal activat
Antenes activades

En aquesta cas la posició màxima a que haurien d'anar les potes del davant és:

límitfísic+2(límitfísic-stbeta)

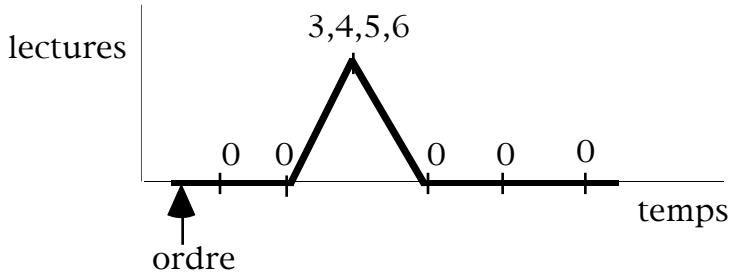
Que és superior al límit lògic fixat. Per evitar arribar a topes caldria ampliar però donat que la situació és excepcional renunciem a tenir un límit lògic tant

Nivell COMPLIANCE

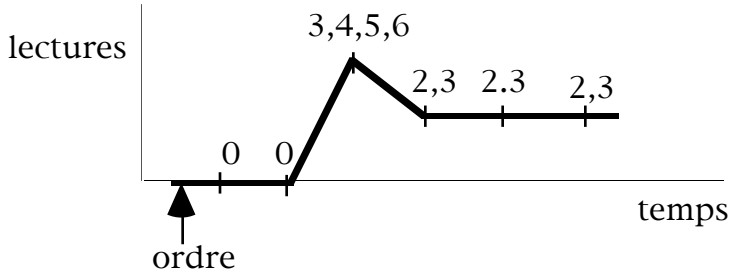
L'objectiu d'aquest nivell és adaptar-se a les peculiaritats de la superficie de
informació de força. El comportament bàsic és el 'No-Vola' que intenta continuam
les potes en contacte amb el terra. Aquest nivell pot èsser desconnectat apujant e
mode. D'aquesta forma podem testejar els nivells inferiors per separat (Hold, Balan

No-vola (0..5)
Entrades:
Força-> Rep, cada 0.02 segons, el consum que el motor beta corresponent e
B-pos-> Posició actual del motor beta corresponent.
Sortides:
Baixa-> Usat per ordenar decendir una posició per apropar-se al terra.
No-bb-> Per inhibir l'acció del beta-balance mentre una pota està baixant o b
dubtós (obrint-se,...).
Estat-> Informa a la resta de comportaments de l'estat en que està la pota. D
hi ha tres estats possibles:
$fent-pas: pota aixecant-se.
$xoc: pota recolzada a terra.
$volant: pota baixant o intentant confirmar un contacte dubtós.

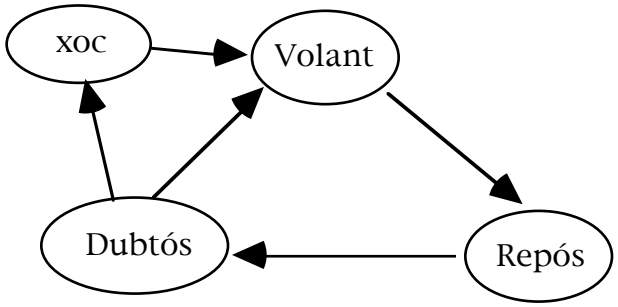
Aquest comportament és el responsable de que totes les potes toquin a terra: c
una pota no hi toca, inicia un descens fins que recupera el contacte (o fins que
Per discriminar si hi ha contacte o no amb el terra s'utilitza el sensor de consu
motor beta. Si durant una estona no es detecta consum se suposa que la pota e
fer-la baixar. El problema que es presenta en aquest descens és que el sens
diferents de zero tot i que realitzem moviments a l'aire (tot moviment repres
Aquest "soroll" no ha de ser confós amb un xoc amb el terra. Les lectu
conseqüència de realitzar un moviment de descens de longitud 1 sense tocar a



encara que de vegades el pic s'allarga una mica més o tarda una mica més
lectures que es produeixen quan hi ha contacte amb el terra són:



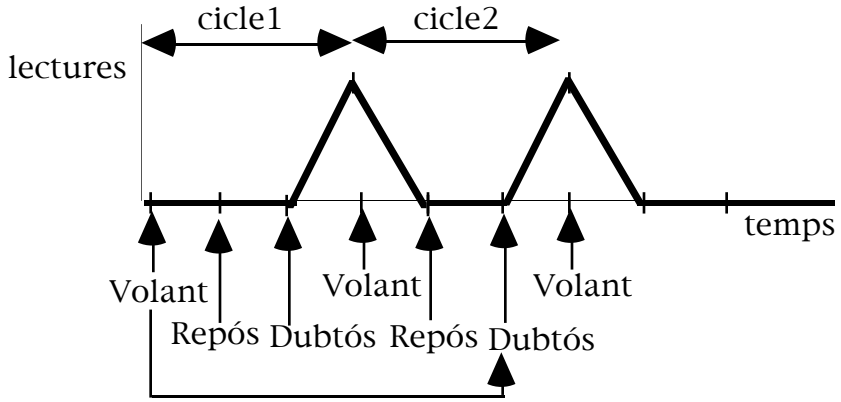
Per saber si un moviment de descens ha provocat contacte amb el terra cald mínim, a la quarta lectura després de enviar l'ordre (si es zero encara no hem continuar baixant). Però, com ja hem dit, molt cops el pic en les lectures s'alla tant la quarta lectura no és, la majoria de les vegades, prou significativa i c cinquena o més. Amb tot això tenim que es pot realitzar un moviment de c lectures de força (o el que ve a ser el mateix cada 0.1 segons ja que el senso lectures cada 0.02 segons). Per tant un descens de 15 posicions (típic en fer u segons en realitzar-se cosa que sembla una mica excesiu.
Per tal d'incrementar aquesta velocitat de descens proposem que el sensor de segons el següent autòmata:



En el que les transicions es realitzen de forma síncrona amb l'arribada d'un força. En cadascun del estats de l'autòmata es fa el següent:

- Xoc: Si es reben suficients lectures amb valor 0 es passa a l'estat Volant
- Volant: Si encara no s'ha arribat a tope, s'ordena a la pota baixar una p a l'estat Repós. En cas d'arribar a tope es poden fer dos coses: manter Volant (cosa que aturarà el robot) o bé passar a l'estat Xoc (cosa que p caminant). En la nostra implementació, els sensors de les potes extreme tipus i els sensors de les potes del mig són de la segona classe.
- Repós: Un cop s'ha rebut una nova lectura de força (que no s'utilitz passa a l'estat Dubtós
- Dubtós: Si la lectura que es rep és superior o igual a un cert llindar (ac 2) es considera que s'ha tocat a terra i es passa a l'estat de Xoc però rec xoc poc segur cosa que farà que poques lectures a zero ja ens facin torna (per tant una confusió del sensor se solventa ràpidament). Si, en canvi, rep en aquest estat és inferior al llindar es passa a l'estat Volant per cont

D'aquesta forma aconseguim realitzar un moviment de descens cada tres lect dir cada 0.06 segons) cosa que representa una millora considerable respecte a presentat abans. Cal mirar-se les lectures de força per entendre realment cor estratègia. En moviments de descens consecutius el perfil de les lectures serà:



Com es veu no tenim més que la superposició de dos moviments simples descrits. Per tant la lectura de força de l'estat Dubtos correspon a la força "ne moviment ordenat en l'estat Volant del cicle anterior. El pic de consum d'un r en un estat Volant cau sobre el següent estat Volant i, si s'allarga, sobre l'esta continuació (en cap dels dos estats es fa cas de la lectura de consum). Per tar Dubtós, la lectura de consum ja es prou indicativa de si hi ha hagut contacte a es que no ja hem ordenat baixar una posició més (en l'estat Volant del segon haurem ordenat un moviment innecessari (en guanyar velocitat perdem precis

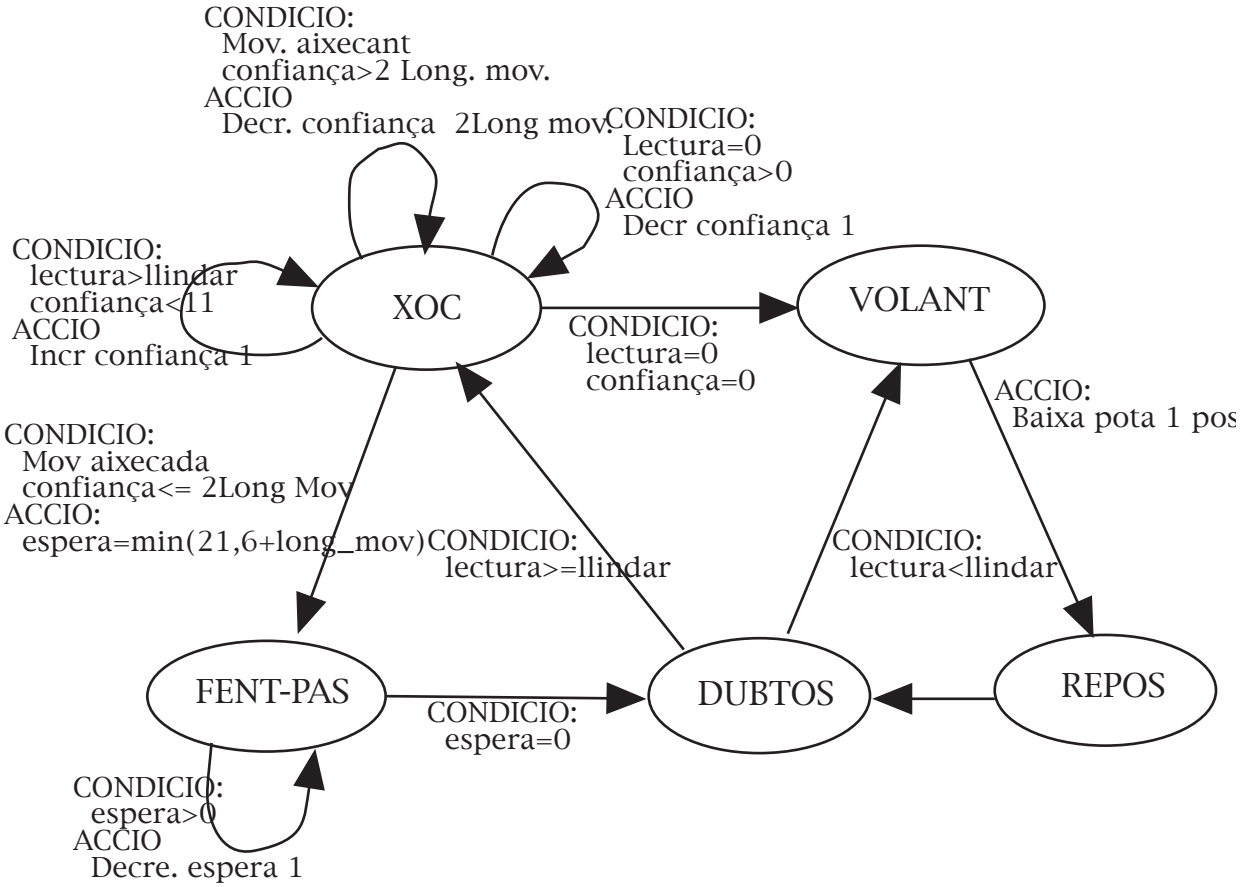
En l'estat de Xoc tenim una variable ('confiança') que indica quina seguretat t hem detectat. Lectures de consum altes (per sobre d'un llindar) incrementen (fins uns màxim) i lectures nul.les la decrementen. Si en decrementar-la es fa com que la pota ja no està en contacte amb el terra i es passa a l'estat Volan d'aquesta variable, fixat en la transició de l'estat Dubtós a Xoc, és petit (2) només 2 lectures consecutives de consum zero ja ens facin tornar a fer baixar contactes es resolen ràpidament. Una altra de les coses que fa disminuïr l contacte és un moviment aixecant una pota. En aquest cas la variable confiar proporcionalment a la llargada del moviment realitzat. Si arriba a ser zero (per o per varis moviments petits consecutius) passem a l'estat Fent-pas en el que de les lectures de consum: el sensor està desconnectat. Després d'aquest pe l'estat Dubtós en el que es decideix si encara estem tocant a terra o no. La quar el sensor es desconnecta a causa d'un moviment d'aixecada ha de ser tal qu realitzi completament el moviment per tal que les lectures causades per aques sensor. En la següent taula es pot veure el que tarden en fer-se diferents movi llargades creixents:

Longitud	Temps
2	5
4	9
6	11
8	11
10	14
12	13
14	15
20	13
30	18
40	18
50	20

El temps està expresat en lectures de força que es poden fer mentre es realitza ja hem dit es pot fer una lectura cada 0.02 segons).
En el programa es calcula el temps d'ascens segons la fòrmula:

(min 21 (+ 6 llargada-moviment))

Que aproxima en excreix els temps de la taula anterior.
L'autòmata complet que s'implementa en el comportament no-vola és:



Una altra de les funcions del comportament no-vola es la d'informar a la res l'estat en beta en que es troba la pota corresponent a través de l'output 'est poden sortir per aquest registre no es corresponen exactament amb els estats acabem de representar. Aquest valors són:

Volant-> Si l'estat de l'autòmata és Volant, Repós, Dubtós o bé Xoc ar menor a 7.
Fent-pas-> Si l'estat de l'autòmata és Fent-pas.
Xoc-> Si l'estat de l'autòmata és Xoc amb confiança major o igual a 7.

A més, en algun d'aquests estats cal evitar que el beta balance pugui interfe evitar que, mentre una pota realitza un descens pas a pas, el beta balance tar ordres (el procés abans descrit per veure si s'ha xocat amb el terra o no queda es basa en monitoritzar moviments de descens unitaris aïllats). Una cosa sim sensor està desconnectat (en l'estat Fent-pas): ordres del beta-balance podrier en que no es pot fer cas de les lectures dels sensors més del previst. Per tot a amb la sortida que indica l'estat en que està la pota, s'envia, quan es precí inhibeix l'acció del beta balance sobre la pota corresponent. Cal anar en con que cada inhibició que ordenem dura uns 0.08 segons (el doble del temps Behavior Language) i que per tant en moments propers a un canvi d'estat j senyal inhibitòria. Per exemple en l'estat Fent-pas s'envia la senyal per inhib sempre que la variable espera és més gran que 2: la senyal enviada en el cicle 3 ja dura fins que sortim de l'estat Dubtós moment en el que reemprendrem anat cap a l'estat Volant) o ja no caldrà mantenir-la (si anem a Xoc).

Podem evitar que aquest comportament funcioni apujant el dip 0 dels dips de
Aturar aquest comportament equival, pràcticament, a desconnectar tot el nivel

Sensor-a(0..5)

Entrades:

Actual-> Posició actual del motor alfa corresponent.

Força-> Cada 0.02 segons rep el valor del consum que actualment fa el motc

Sortides:

Estat-> Estat en el que es troba la el motor alfa. Pot ser:

Movent-> El motor està realitzant un moviment.

Xoc-> El moviment ha acabat en un xoc amb un obstacle.

Repós-> El moviment ha acabat sense xoc.

Aquest comportament monitoritza els moviments ordenats al motor alfa que li de detectar l'inici d'un moviment llarg (major a la constant $límit-alfa) és pass durant el qual es monitoritzen les forces que fa el motor. Si es detecten més de un determinat llindar es passa immediatament a l'estat de Xoc. Si s'acaba el detectar forces elevades es passa a l'estat de Repós. Per saber quanta estc monitoritzant les forces per a cada moviment es fa un càlcul a partir de la s següent taula mostra un exemple de quantes lectures de força es poden fer per de diverses longituds:

Longitud	Temps
5	9
10	10
15	11
20	11
25	14
30	14
35	18
40	16
45	21
50	22
55	20

En el programa s'aproxima (sempre en escreix per estar més segurs de que acabat) aquest nombre de lectures segons la fòrmula:

$$(long_mov/2)+7$$

Limitant el resultat a l'interval [11,25].

La sensibilitat d'aquest sensor per als moviments en alfa es pot ajustar simple paràmetres:

El llindar de força.

El nombre de lectures per sobre del llindar per ser considerat Xoc.

Salta(0..5)

Entrades:
Estat-> Estat del motor alfa d'aquesta pota subministrat pel comportament a
B-pos-> Posició en beta de la pota.
A-pos-> Posició en alfa de la pota.
Sortides:
Amove-> Moviment en alfa.
Bmove-> Moviment en beta.

Aquest comportament vigila els moviments que es fan amb el motor alfa de la
i si un d'ells provoca un xoc (indicat pel comportament anteriorment descrit)
el que anomenarem rebot:

1.- Un moviment en el sentit contrari al que ha provocat el xoc i de la ma
2.- Aixecar la pota una mica (unes 15 posicions).
3.- Un moviment endavant igual al que havia produït el xoc.

Amb aquest reflex es preten poder superar un obstacle que es troba en
aconseguir posar la pota sobre ell (fent, si cal, més d'un rebot consecutiu).
En cas de que la pota estigui massa alta com per aixecar-la encara més llavors
en el 3, el moviment endavant, es fa la meitat de llarg que el que havia provo
encara dispara el rebot llavors es tornarà a dividir per dos la llargada (es a
quarta part que el moviment original) i així succesivament fins que no es t
rebot. Això es fa amb l'intenció d'apropar la pota a l'obstacle que tant costa de

Pas(0..5)
Entrades:
Senyal-> Quan es rep s'inicia el procés per a fer un pas.
B-pos-> Posició del motor beta de la pota corresponent.
Llargada-> Desplaçament sobre el registre zero amb el que s'ha de fer el pas
del nivell DRIVE.
Alçada-> Alçada amb la que, inicialment, s'ha de fer el pas.
Estat-> Estat en beta de la pota corresponent proporcionat pel comportamer
Força-> Força que, en cada moment, està fent el motor beta.
Zero-> Zero actual del comportament alfa balance proporcionat pel comport
nivell ADAPTACIO.
A-pos-> Posició del motor alfa de la pota corresponent.
Sortides:
Amove-> Moviment en alfa.
Bmove-> Moviment en beta.

Quan es rep un missatge en el registre 'senyal' s'executa un pas. Un pas consi
pota fins que no se sent força i en tirar-la endavant (o endarrera) fins a la p
(zero+llargada).

Abans de començar tot el procés de fer el pas emmagatzemem en registres te
de 'llargada' i 'alçada' per tal que si algú els modifica mentre fem el pas no ens

Un cop fet això passem a executar el pas. Això compren:

1.- Aixecar la pota

2.- Monitoritzar les forces mentre el comportament no-vola indica que
'Fent-pas'. Si quan s'ha rebut l'ordre de fer el pas algú està inhi
comportament, el punt 1 no s'haurà portat a la pràctica i per tant el co
vola no passarà a l'estat 'fent-pas'. Cal tenir això en compte per no que
infinit (cal posar, per exemple, un temps màxim d'espera d'entrada e
pas').
3.- Si durant el periode en el que es fa el pas al menys una tercera pa
llegides ha estat inferior a un determinat llindar es considera que la pot
per tant s'ordena el moviment d'avançar
Si no es així es torna al punt 1 però fixant l'alçada a valor 5 (se suposa
per deixar el contacte amb el terra i que per tant no cal apujar gaire
forma aconseguim un moviment de relaxació progressiva de la pota
transferint poc a poc, la força que suporta a les seves veines. Així ev
sacsejades força desagradables (sobretot en baixar dels obstacles).

En aquest procés hi ha tres casos en els que s'aborta un pas que són:

- Abans de començar el pas la pota ja és a la posició indicada per (zero+l
més enllà d'aquesta posició.
- Si abans d'aixecar la pota (punt 1) es veu que ja està massa alta.
- Si es veu que el fet d'aixecar la pota ha provocat un increment de la fo
que es detectava al principi del pas i no una disminució (com seria d
s'interpreta com que la pota ha quedat atrapada en un forat i es rer
endavant (el que es fa es tirar-la avall per desfer el moviment d'aixecada
l'engantxada). Es confia que els moviments de les altres potes posaran
millor postura per a la propera vegada en que s'intenti de fer el pas.

Nivell CLIFF

Darrer nivell de la part d'adaptació a les irregularitats del terreny. La seva miss potes extremes del robot quedin sense tocar a terra. La idea es que si, ni amb l'ac es proporciona la diferència límit físic-límit lògic hem aconseguit tocar a terra, cal l'acomodació per apropar-se al fons del forat en el que hem posat la pota i explora a la búsqueda d'un punt on recolzar-la. Aquest nivell també es desconnecta quan mode está apujat.

Irs

Entrades:

Inclinació-> Usem l'informació que proporciona el comportament 'penc ADAPTATION per saber l'inclinació actual del robot. Si el robot està aixe no cal fer cas de les lectures nul.les del sensors d'infrarojos del davant .

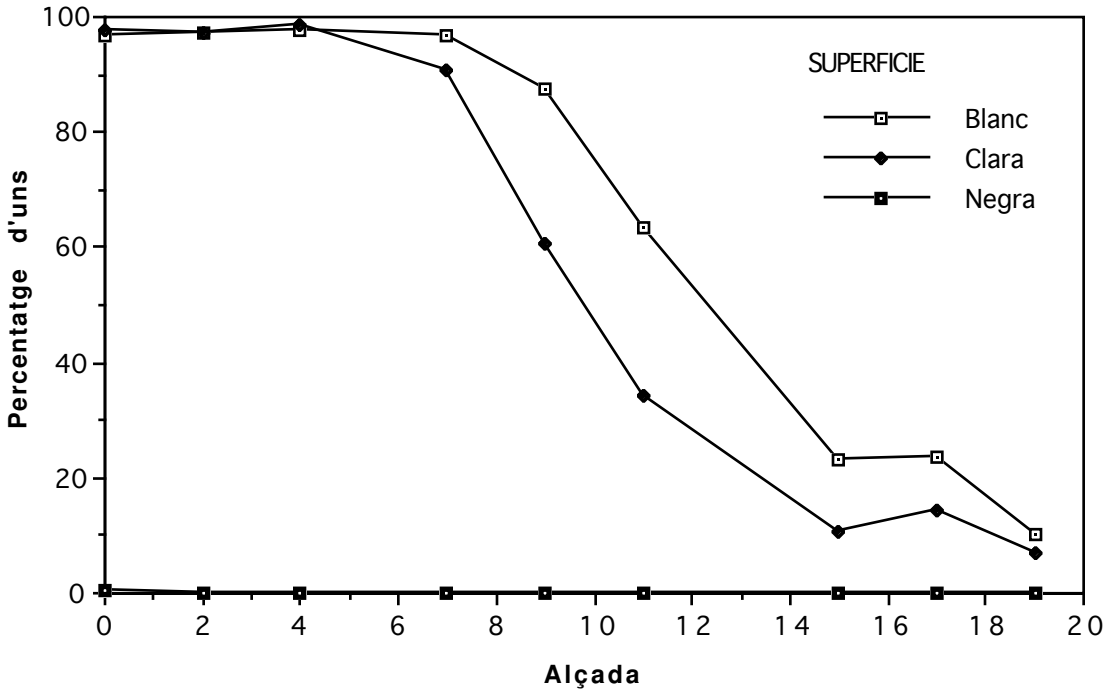
Sortides:

Terra-> Val 1 si es detecta terra amb els sensors d'infraroig i 0 en cas contra

Aquest comportament utilitza els sensors d'infrarojos montats sobre les quat del robot per a veure si hi ha una superficie propera al robot a l'area en que un lloc per recolzar-se.

Originariament els sensors d'infrarojos estaven orientats horitzontalment però els hem girat 90 graus i estan apuntant al terra (en realitat fins ara només sensor que està sobre la pota zero però caldria fer-ho per tots i replicar aques cadascun d'ells).

El gran problema que donen els sensors d'infraroig és que són molt sens superfícies sobre la que s'utilitzen (en concret al color de la superfície) tal i següent gràfica. En ella s'indica el tant per cent de lectures a 1 que dona el s col.locat sobre diferents superfícies amb diferents alçades (l'alçada es dona partir de la posició stantbeta):



Sembla clar que com més fosca és la superfície menys útils són els infrarojo
problema és realment greu pel fet que en l'entorn en que es mou el robot ter
diversos colors.
Tots aquests problemes ens porten a restringir l'ús dels sensors d'infrarojc
detecta terra o no, prescindint d'intentar determinar a quina distància es t
corrent el risc de que, en algunes circunstàncies el robot pugui patir "enga
superficie fosca li pot fer pensar que hi ha un precipici al davant cosa que, qua
actuiï, el farà retrocedir i girar).

En el comportament IRS es determina la presència o no de terra de la següent f
Es llegeix 20 cops el sensor d'infraroig (més o menys durant un segon) i si hi h
de lectures a 1 es considera que hi ha terra. Si durant 5 segons seguits no s'h
del 25% de lectures a 1 llavors es determina que hi ha un forat sota el sensor
decisió està supeditada a l'informació que ens dona l'inclinòmetre: si aquest
robot està aixecat del davant, apuntant amb el sensor d'infrarojos en diagc
llavors es considera l'informació sensor sense validesa i la sortida Terra val 1.
ha una superfície propera a la pota però que no l'estem veient perque el robo
altres comportaments d'aquest nivell s'encarregaran d'orientar-lo correctamen
fer cas del sensor.

Explora-davant(0,1)
Entrades:
B-pos-> Posició del motor beta de la pota corresponent.
Terra-> Indicació de si el sensor IRS detecta terra o no. Ara per ara s'utilitza
dues potes del davant però caldria fer que cada pota utilitzés el seu.
Estat-> Estat de la pota en beta.
A-pos-> Posició del motor alfa.
Sortides:
A-delta-> Ordres al motor alfa.
B-delta-> Ordres al motor beta.
Inhibir-pas-> Sortida per evitar fer un pas mentre estem explorant el terreny

Quan es detecta que una pota del davant ha arribat al límit lògic sense tocar a
búsqueda d'algun lloc on recolzar-la. Aquesta búsqueda es basa en aixe
desplaçant-la a diferents posicions en alfa per després deixar que el comporta
fent-la baixar. Si el registre terra val 1 (es detecta la presència d'una superfície
les posicions en alfa que es visiten són tant endavant com endarrera segons la

- Anar a la posició zero.
- Anar alternativament endavant i endarrera amb amplituds creixents.

Si no es detecta terra el que es fa és moviments només endarrera amb l'esper
mateixa superficie sobre la que estan les potes del mig. Si tot va bé, s'acaba
penjada just al caire del precipici.

Mentre s'executen aquests moviments d'exploració cal tenir inhibida la possibi
per tal de que aquest no interfereixi en la seqüència de moviments en alfa que

Si el dip 0 dels dips de mode està apujat, no s'inicia la seqüència de movim
cosa que atura tots els comportaments d'aquest nivell.

Explora-darrera(4,5)

Entrades:

B-pos-> Posició del motor beta de la pota corresponent.

Estat-> Estat en beta de la pota

A-pos-> Posició del motor alfa.

Sortides:

Adelta-> Moviments en alfa.

Bdelta-> Moviments en beta.

Inhibir-pas-> Per aturar el comportament 'pas' corresponent.

Es un comportament anàleg a l'anterior però dedicat a les potes posterio
sensors d'infraroig orientats cap a terra en la part posterior del robot fan que
de la dada que indica si es detecta o no una superfície propera. Això fa q
distinguir entre dos estratègies d'exploració. El que es fa es utilitzar sempr
busca en totes direccions.

Aplana-davant

Entrades:

Penjada-davant-> Rep una senyal cada cop que una de les potes del d
moviment causat pels comportaments explora-davant(0,1).

Zero-> Obre l'accés indirecte a l'alçada mitjana del robot per altres comport

Sortides:

Nou-stb-> Nova alçada fixada conjuntament per aquest comportament i
accedeixen al registre d'entrada 'zero'.

Quan es detecta que una de les potes del davant no toca a terra es disminuei
per tal de facilitar la trobada d'un punt de recolzament. Indirectament el c
guanyar molta estabilitat cosa que evita que el robot caigui pels precipicis.
La quantitat en que aquest comportament influeix en l'alçada del robot està lin

$$-\frac{límitfísic - stbeta}{2}$$

que és la meitat del que la podria fer disminuïr. Aquesta limitació té dos raons

- En cas que en abaixar el cos es detecti xoc amb el sensor de tacte de la
el descens i fins i tot pujar una mica. Això s'aconsegueix gràcies a que e
"aixeca" del nivell ADAPTACIO té més capacitat d'influència en l'alçada g
que no pas Aplana-davant

- Cal reservar una part del que el robot pot baixar pel cas en que tam
alguna pota del darrera.

Quan es detecta que una pota està al límit lògic sense sentir força llavors
disminuïr l'alçada del robot a un ritme de 1 punt cada mig segon. La regulaci
descens depen de quants moviments d'exploració volem que faci la pota aban
s'hagi aplanat del tot. Cada moviment d'exploració tarda en realitzar-se:

$$temps_exploració = temps_pujar + temps_baixar =$$
$$= [\min(15 + 6 \quad 21)] \cdot 0.02 + [[15 + (límitlògic - límitfísic)] \cdot 3] \cdot 0.02 =$$
$$= [66 + 3(límitlògic - límitfísic)] \cdot 0.02 \quad (segons)$$
(on 15 és l'alçada del moviments d'exploració)

Que en el nostre cas venen a ser uns 2.5 segons. Si volem que el descens del ro cal que el monostable 'inercia' d'aquest comportament sigui superior al t activacions del comportament explora corresponent (en el nostre cas val 3 seg En aquests 2.5 segons el robot baixarà uns 5 punts, que no són tots els que pc haurà relitzat una exploració abans d'aplanar-se del tot (cosa que es molts cas de continar baixant).
Com es pot observar hi ha una assimetria entre la velocitat a la que es baixa (2 a la que es puja (8 punts cada segon). Això es fa per afavorir una ràpida r postura normal un cop trobat un punt de recolzament.

Aboca-davant
Entrades:
Penjada-davant-> Rep una senyal cada cop que una de les potes del da moviment causat pels comportaments explora-davant(0,1).
Terra-> Detecció o no de terra amb els infrarojos.
Inclinació-> Inclinació actual del robot segons el comportament 'pende ADAPTACIO.
Zero-> Obre l'accés a altres comportaments per actuar sobre l'equilibri front
Sortides:
Nou-bf-> Aportació d'aquest comportament i dels que envien dades al re 'zero' a l'equilibri frontal/posterior del robot.

Aquest comportament complementa a l'anterior en l'intent d'apropar una pota toca a terra a una superfície amb la que fer contacte al fons del forat on s'ha p fa es inclinar el robot (aixecant-lo del darrera i abaixant-lo del davant) a tra l'equilibir frontal/posterior mantingut pel beta balance. El màxim d'aques calculada en el comportament 'aboca' del nivell ADAPTACIO):

2 (límitfísic-stbeta)

Les condicions per disparar aquest comportament són les de tenir alguna pc posició indicada per la constant $límit-soft, no tocar a terra, detectar una supe els sensors d'infrarojos i no estar aplanats del davant per sobre d'un determi disparat ordena incrementar l'inclinació a un ritme d'unes 8 posicions per permet, igual que en el comportament anterior, realitzar un moviment d'e d'acabar d'inclinar-nos del tot. També igual que abans, cal assignar al monos temps superior al que tarda cada moviment d'exploració si volem aconsegui forma continua i monòtona.
El temps en que es recupera l'inclinació inicial, un cop desaparegudes les cond disparat aquest comportament, és molt més ràpid que el temps emprat en incli

Aplana-darrera
Entrades:

Penjada-darrera-> Rep una senyal cada cop que una de les potes del d
moviment causat pels comportaments explora-davant(0,1).

Sortides:

Nou-stb-> Nova alçada mitjana fixada per aquest comportament i els qu
registre 'zero'.

Comportament absolutament anàleg al 'aplana-davant' abans explicat però di
del darrera.

Aboca-darrera

Entrades:

Penjada-darrera-> Rep una senyal cada cop que una de les potes del d
moviment causat pels comportaments explora-davant(0,1).

Inclinació-> Inclinació fontal/posterior actual del robot indicada pel compo
del nivell ADAPTACIO.

Sortides:

Nou-bf-> Nou equilibri frontal definit per aquest comportament.

Comportament anàleg al 'aboca-davant' abans explicat però que actua per les
L'única diferència important està en que, ara com ara, no podem fer ús de ca
infrarojos del darrera del robot per tant la condició de detectar una superfíci
se inclinar que hi havia en el comportament 'aboca-davant' no es té en compte

Nivell WALK

S'encarrega de generar la seqüència de passos adequada per fer avançar al robot millor: sempre s'intenta generar el patró més proper al tripot gait que les conc permenten.

Atura-a-balance(0..5)
Entrades:
Estat-> Estat en beta de la pota corresponent.
Sortides:
Atura-ab-> Substitueix les ordres que l'alfa balance envia a la pota correspor

Quan una pota no toca a terra (estat en beta diferent de Xoc) no té cap pròf l'alfa balance: l'alfa balance que aniria a aquesta pota no es traduïria en avanç toca a terra no es pot empentar de forma efectiva). Aquest comportament det no toca a terra i, en aquest cas, substitueix (suppress) les ordres que envia l pota corresponent per la senyal atura-ab (que té sempre valor 0).

Si el dip 1 dels dips de mode està apujat aquest comportament no actua caminem, no val la pena optimitzar el repartiment del alfa-balance.

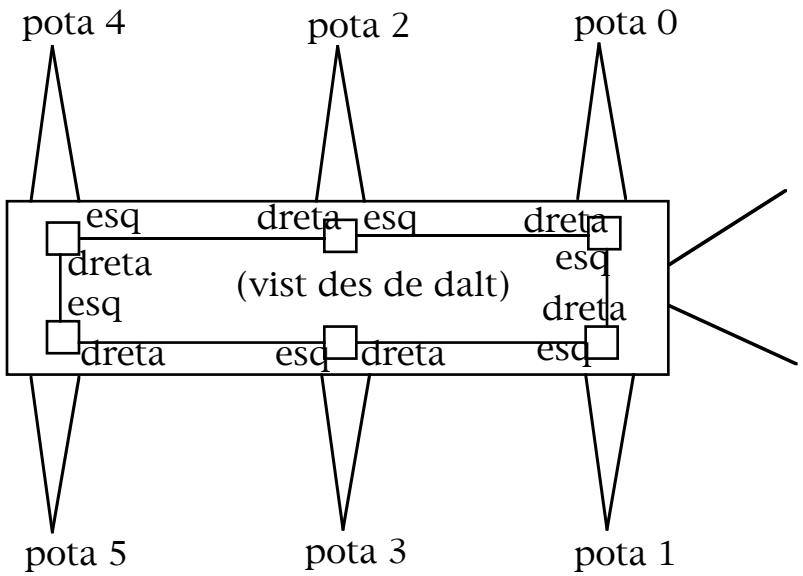
Regula-alçada(0..5)
Entrades:
Xoc-> Estat en alfa de la pota vigilada.
Seguent-pas-> Es rep una senyal cada cop que la pota vigilada fa un pas.
Sortides:
Nova-alçada-> Canvi respecte a l'alçada standard per als passos fets per la p

Per tal de maximitzar la velocitat d'avanç del robot s'ha programat el compor una alçada del pas per defecte molt baixa. D'aquesta forma cada pas s'executa L'inconvenient que té això és que, en passar per una zona accidentada continuament disparant el rebot ja que amb l'alçada per defecte no n'hi ha pro petit bony del terreny. Aquest comportament intenta evitar això: un cop c s'incrementa l'alçada amb que es fan els dos següents passos de la pota qu l'obstacle. Així si hi ha algun altre petit obstacle se superarà a la primera.

Torn(0..5)
Entrades:
Senyal-dreta-> Es rep una senyal si la pota que hi ha a la dreta ens cedeix el pas.
Senyal-esq-> Cessió de torn per part de la pota de l'esquerra.
Estatb-esq-> Estat en beta de la pota de l'esquerra.
Estatb-dreta-> Estat en beta de la pota de la dreta.
Estata-esq-> Estat en alfa de la pota de l'esquerra.
Estata-dreta-> Estat en alfa de la pota de la dreta.
Canvi-> Si es rep una senyal en aquest registre, s'inverteixen els torns (zer l'inrevés).
Sortides:

Go-> Ordre al comportament 'pas' de que s'executi i cessió de torn a les pot

Aquests 6 comportaments generen, de forma distribuïda, el patró de passos
Genghis. Quan un comportament Torn ha rebut el permís (el torn, escrit a
distiguir-lo del nom del comportament) dels dos comportaments Torn veïns (
i les potes veïnes estan en la situació normal per a una pota (Xoc en beta i Re
pot ordenar que s'executi el pas de la pota corresponent i lliurar el torn
relacions de veïnatge es defineixen de la següent forma:



El fet de coneixer quina pota té el torn en cada moment dins de les relacio
establert entre potes veïnes, ens permet saber en quin ordre es faran els p
permeten establir una relació d'ordre temporal entre les potes. Les tres poss

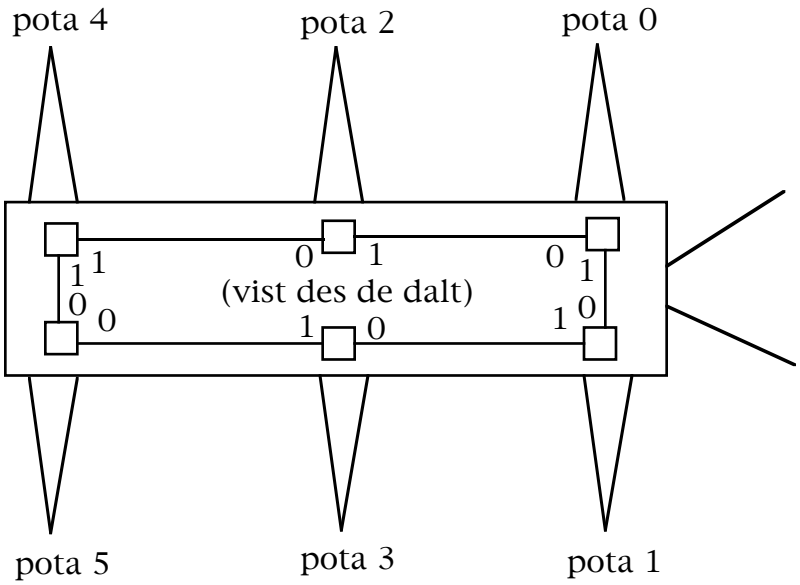
- Una pota té un torn però no l'altre (per exemple, té el torn que ve d
no el de la dreta): Vol dir que la pota de l'esquerra ja ha fet el seu pa
encara està per fer-lo. Dit d'un altra forma: la pota en qüestió farà el
que la de l'esquerra però després que la de la dreta.
- Una pota té els dos torns: Està a punt de fer el pas. Les veïnes estan
faci per recuperar el torn. Farà el pas abans que les dos veïnes. Es un
dos casos com l'anterior.
- Una pota encara ha de rebre els dos torns: Tot just acaba de fer el p
fer-lo les veïnes. Les dos veïnes han de fer el pas abans aquesta el tor

Amb aquesta semàntica es veuen clares certes restriccions en el repartiment

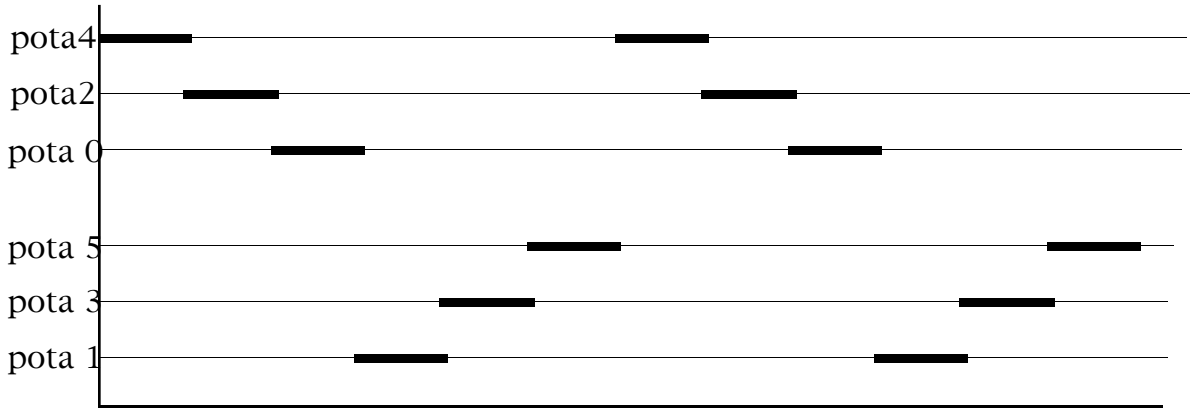
- En tot moment hi ha d'haver al menys una pota amb dos torns i una
l'anterior) sense cap d'ells.
- No pot ser que totes les potes tinguin només un torn (implicaria una esp
- Dos potes veïnes no poden posseïr el mateix torn alhora

.....

També a partir d'això es desprén que, donada una succesió cíclica de passo
l'ordre d'aquesta successió simplement canviant les relacions temporals de
Es a dir, canviant els torns de mans invertim l'ordre en que es fan els passo
aquest proposit es per al que tenim l'input 'canvi' que, en rebre's simultan
comportaments Torn inverteix les relacions temporals definides. Si això bé
una inversió de la longitud del pas, llavors posarem el Genghis a desfer el c
Segons com s'inicialitzin els torns en cada comportament Torn podem aco
patrons de passos. Per exemple, per una inicialització:

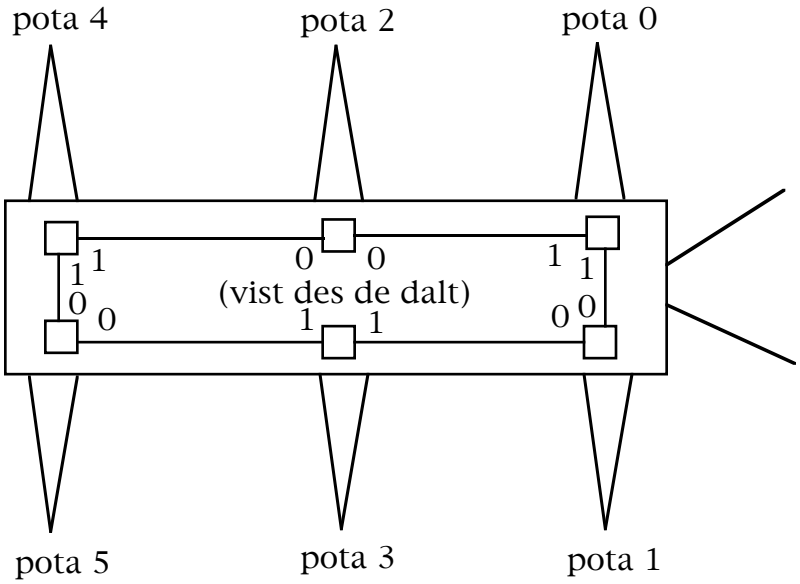


Aconseguim caminar tal com:

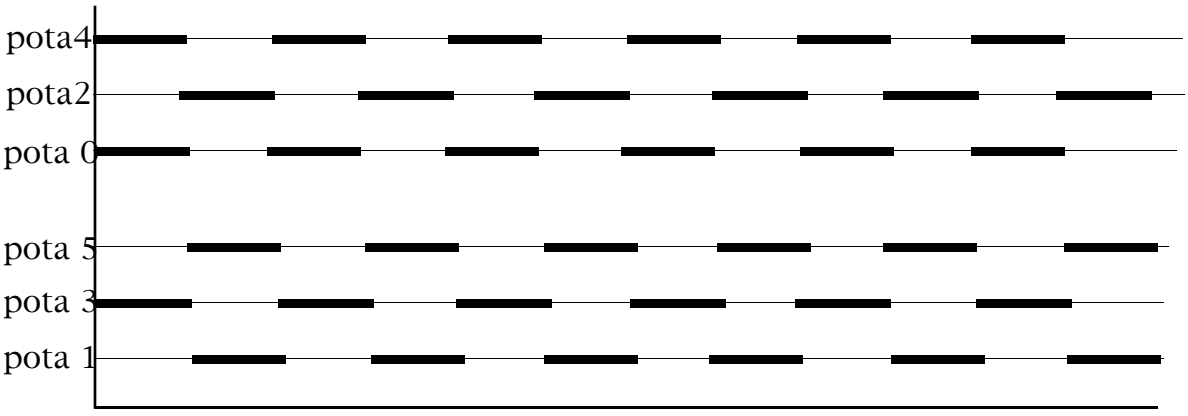


que no és metacronal (les ones d'un costat i de l'altre van en sentits contrari

Amb una inicialització com la nostra:



El patró resultant és el tripot gait (és el patró que s'hauria d'observar e
condicions del terreny ho permeten):



Primer-pas

Sortides:

Go-> Inicialitza alguns torn

En el nostre cas l'inicialització els torns es produeix en dos parts. La primera
dels comportaments Torn en la que es deixen per assignar alguns valors (els
La segona es des d'aquest comportament en el que, si el dip de mode número
esquerra del robot) està abaixat, espera dos segons i envia els valors que falta
el robot a caminar. D'aquesta forma podem engegar el robot i veure com fun
quan ja en tinguem prou, abaixar el DIP 0 i deixar que el Genghis comenci a ca

Nivell DRIVE

Fins al nivell anterior s'aconsegueix fer que el Genghis camini però sempre endav
subsumeix a tots els anteriors i aconsegueix controlar la direcció i velocitat de l'ava
events que fan que es canviïn aquests paràmetres són:

- Cal evitar xocar contra obstacles: s'utilitzen les antenes per detectar obstac
es manté una estona es gira en la direcció contraria a la de l'antena que detec
les dos antenes les que el detecten es tira endarrera i després es gira a l'esque
- Cal evitar caure en els precipicis: si una pota del davant o del darrera del r
trobar terra s'interpreta com que és en un precipici i es desencadena una es
se'n.
- A efectes de testejar el programa, s'ha afegit un comportament que fa que el
sempre a pujar. Així, si detecta que cap a la banda esquerra hi ha més bonys
dreta, girarà poc a poc a l'esquerra. D'aquesta forma s'evita la tendència natu
anar cap als llocs més baixos (provocada en part per tenir "rodes" a l'extrem
part perque quan una pota està més alta té més recorregut en alfa cosa que f
al costat de les potes en mitjana més baixes).

L'efecte de gir global no es més que l'acumulació de girs provocats per cadascu
anteriors.
La forma en que es controla la velocitat de robot és a base de disminuir la llargad
fa. En el cas extrem de voler anar endarrera cal posar llargades de pas negatives.
En canvi, la direcció es controla per la diferència de longituds de pas entre la banda
dreta del robot. Així si la banda esquerra té una longitud de pas més gran que la
estarem fent girar el Genghis cap a la dreta. Els casos extrems de gir es correspon
punt fix moment en el que la longitud de pas d'una banda del robot és igual a la
signe contrari.
En resum podem dir que:

$$velocitat_avanç = long_pas_esq + long_pas_dreta$$

on positiu vol dir endavant i negatiu endarrera. I també tenim que:

$$direcció_avanç = long_pas_esq - long_pas_dreta$$

on negatiu vol dir gir a l'esquerra i positiu a la dreta.
Aquest nivell no té cap més efecte a part de la modificació de les llargades de pas
estem caminant (dip 1 dels dips de mode apujat) els seus efectes no es noten.

Ara passem a comentar els comportaments que constitueixem aquest nivell.

Canvi-pas

Entrades:

ve0-> Influència del comportament xoc en la longitud de pas de la banda esc
vd0-> Influència del comportament xoc en la longitud de pas de la banda dr
ve1-> Influència del comportament precipicis-davant en la longitud de p
esquerra.
vd1-> Influència del comportament precipicis-davant en la longitud de pas d
ve2-> Influència del comportament precipicis-darrera en la longitud de
esquerra.
vd2-> Influència del comportament precipicis-darrera en la longitud de pas
ve3-> Influència del comportament obstacles-esq en la longitud de pas de la
vd3-> Influència del comportament obstacles-esq en la longitud de pas de la

ve4-> Influència del comportament obstacles-dreta en la longitud de pas de
vd4-> Influència del comportament obstacles-dreta en la longitud de pas de

Sortides:

pas-e-> Longitud definitiva dels passos de la banda esquerra del robot.
pas-d-> El mateix per la banda dreta.

Aquest comportament centralitza la modificació de les longituds dels passos d
i dreta del robot. En realitat no fa més que sumar les influències dels diferer
que pretenen controlar el robot al mateix temps.
La longitud per defecte (quan cap dels comportament d'aquest nivell està act
per la constant $pas. Tries adequades d'aquesta constant (en funció de divers
ara la distància entre eixos de les potes,...) ens poden acostar a estratègies co
"Follow the Leader". El problema es que aquestes tries són dependents del pa
estem fent i això, en el nostre cas, no es fixe sino que ve determinat per le
terreny. Valors raonables per aquesta constant són aquells que proporcionen
(stroke) dins l'interval:

$$\left[\frac{1}{3}\text{distància_eixos}, \frac{1}{2}\text{distància_eixos}\right]$$

Xoc

Entrades:

pas-e-> Rep un tic per cada pas que fa la banda esquerra del robot.
pas-d-> El mateix per la banda dreta.

Sortides:

pas-esq-> Aportació d'aquest comportament a la longitud dels passos de la
pas-dret-> Aportació a la longitud dels passos de la banda dreta.

Aquest comportament implementa una primitiva estratègia per evitar ob
consisteix girar en el sentit contrari al dels obstacles detectats per les antene
es detecta un obstacle amb l'antena esquerra, s'ha de girar a la dreta (cosa
decrementant la llargada dels passos de la banda dreta del robot). Per l'ant
cosa simètrica. Si són les dos antenes les que detecten contacte, llavors el mo
composar els dos girs fa retrocedir el robot.
Els girs a esquerra i dreta es fan amb una certa inèrcia (es mantenen durant u
la banda a la que s'ha modificat la longitud de pas) per assegurar-se d'haber
un costat. La inèrcia del gir a l'esquerra és major que la del gir a la dreta.
s'aconsegueix que, en cas de detectar contacte amb les dos antenes, després d
produeixi un lleuger gir a l'esquerra. Així s'evita que, en tornar endavant alt
xoc amb el mateix obstacle que abans amb la mateixa orientació.

Precipicis-davant

Entrades:

Amplitud0-> Amplitud amb que la pota 0 está realitzant moviments explora
Amplitud1-> El mateix per la pota 1.
Pas-fet-> Compta passos fets ja sigui per la pota 0 o per la 1.
Pas-desfet-> Compta els rebots d'aquestes dos potes. Un pas seguit d'un r
encara no hem sortit del tot del precipici.
Estat0-> Estat en beta de la pota 0.
Estat1-> Estat en beta de la pota 1.

Estat-a0-> Estat en alfa de la pota 0.
Estat-a1-> Estat en alfa de la pota 1.
Terra-> Detecció o no del terra per part dels infrarojos.
E4-> Ens indica si la pota 4 està o no en un precipici.
E5-> El mateix per la pota 5.

Sortides:

Pas-esq-> Aportació d'aquest comportament a la longitud del pas de la band
Pas-dret-> El mateix per la banda dreta.
Fals-estat0-> Quan la pota 0 està en un precipici cal facilitar el moviment de per a poder tirar endarrera (altrament, una pota penjada bloqueja la ma aquesta raó s'enganya els comportaments Torn de les potes 1 i 2 subm falsos estats en beta i en alfa de la pota 0 i indicant-els-hi continuament q pota 0, tenen el permís (torn) per a fer el pas quan vulgin (es a dir, non coordinin amb l'altra veïna). Aquesta senyal és el fals estat en beta de l Torns 1 i 2.
Fals-estat1-> En cas d'estar penjada la pota 1 també cal enganyar a les veïn la 0 i la 3). Aquesta senyal és el fals estat en beta de la pota 1. Si coindeix del davant en un precipici llavors les potes del mig prescindiran de les del caminarà endarrera utilitzant només les quatre potes posteriors.
Falsa-senyal0-> Senyal utilitzada per indicar continuament a les potes 1 i permís de la pota 0 per a fer el pas.
Falsa-senyal1-> El mateix però ara es la pota 1 que ho diu a les potes 0 i 3.
Fals estata0-> Fals estat en alfa de la pota 0 que arriba als Torns 1 i 2.
Fals-estata1-> Fals estat en alfa de la pota 1 per als Torns 0 i 3.
E0-> Indica, a altres comportaments, l'estat de la pota zero respecte als pre que no es en cap precipici i 1 vol dir que si.
E1-> El mateix per la pota 1.

Quan una pota del davant explora durant força estona[1] sense trobar cap punt quan els infrarojos no detecten el terra llavors cal suposar que el robot e precipici. En aquest cas és quan es desperta aquest comportament l'objectiu de el robot d'aquest tipus de situacions crítiques. Les accions que s'emprenen són

1.- Intentar fer pujar la pota que és al precipici a terreny segur
2.- Retrocedir uns passos
3.- Girar cap a l'esquerra uns 90 graus

El primer punt s'intenta assolir de diferents formes segons la pota penjada

- Si és la dreta: es retrocedeix mentre es gira a la dreta per aproximar la caire del precipici.
- Si es l'esquerra: es retrocedeix mentre es gira a l'esquerra.
- Si són les dos: es retrocedeix sense girar.

Mentre es realitzen aquestes moviments la pota (o potes) penjada continua fe búsqueda d'un superfície on recolzar-se (per intervenció dels comportaments

[1] El temps que fa que una pota està explorant es dedueix a partir de l'amplitud d d'exploració que realitza (com més grans són més temps fa). Es pot regular el grau d Genghis disminuïnt o bé augmentant l'amplitud a partir de la que es dispara el d'evitar precipicis.

CLIFF). Quan sigui prou aprop del caire del precipici on s'ha posat, hi xocarà, el reflex rebot que acabarà posat la pota al caire del precipici.
Quan ja s'ha recolzat la pota en una posició segura, es retrocedeix uns pa compten com a passos bons aquells que no van seguits d'un rebot. així s'evi aprop del precipi a causa de que un contacte amb el caire del precipici ha esta pota ja estava a terreny segur. Després de retrocedir, es gira uns 90 graus allunyar-nos de la direcció en que hi havia el precipici. Si aquest gir es fa en la a la de la pota que ha detectat el buit llavors el Genghis acaba indefectiblemen Es per això que hem optat per fer els girs sempre en el mateix sentit.
Com hem mencionat, els sensors d'infrarojos també poden desencadenar aqu fugida dels precipicis. Això porta a que, caminar sobre una superfície ne considerat com un perill i el Genghis maniobra per evitar-la.
Si mentre s'està executant l'estratègia per salvar un precipici detectat per les p potes del darrera també en detecten un llavors es resetaja tot el seguit de pass a l'esquerra que es volien fer (ja no té sentit fer-los).

Precipicis-darrera

Entrades:

- Estat4-> Estat en beta de la pota 4.
- Estat5-> Estat en beta de la pota 5.
- Amplitud4-> Amplitud amb que la pota 4 está realitzant moviments cer recolzament.
- Amplitud5-> El mateix per la pota 5.
- Estat-a4-> Estat en alfa de la pota 4.
- Estat-a5-> Estat en alfa de la pota 5.
- Pas-fet-> Compta passos fets ja sigui per la pota 4 o per la 5.
- Pas-desfet-> Compta els rebots d'aquestes dos potes. Un pas seguit d'un r encara no hem sortit del tot del precipici.
- E0-> Ens indica si la pota 0 està o no en un precipici.
- E1-> El mateix per la pota 1.

Sortides:

- Vp-e-> Aportació d'aquest comportament a la longitiut del pas de la banda e
- Vp-d-> El mateix per la banda dreta
- Falsa-t4-> Senyal que indica continuament a les potes 2 i 5 que tenen el pe per a fer el pas. S'utilitza quan la pota 4 està penjada en un precipici.
- Falsa-t5-> El mateix però ara es la pota 5 qui ho diu a les potes 3 i 4.
- Fals-e4-> Aquesta senyal és el fals estat en beta de la pota 4 envia cap al permetre el seu funcionament quan la pota 4 està en un precipici..
- Fals-e5-> En cas d'estar penjada la pota 5 també cal enganyar a les veïnes (i la 5).
- Fals ea4-> Fals estat en alfa de la pota 4 que arriba als torns 2 i 5.
- Fals-ea5-> Fals estat en alfa de la pota 5 per als torns 3 i 4.
- E4-> Indica l'estat de la pota 4: 0 vol dir que no es en cap precipici i 1 vol d
- E5-> El mateix per la pota 5.

Comportament anàleg a l'anterior però dedicat a les potes del darrera. L'unica es que no pot utilitzar informació d'infrarojos per activar-se. Per compensar aquesta mancança hem fet que el Genghis sigui més covard per als precipicis

endarrera: li costa menys decidir que no val la pena continuar buscant un pu
deseguida desencadena l'estratègia de salvar-se del possible precipici.
Quan coincideixen potes del davant i potes del darrera en un precipici és un d
que es pot trobar el Genghis. Les influències d'aquest comportament i les
combinen de tal forma que el moviment net és sempre el d'intentar fer pujar l
una superfície segura per després concentar-se en l'estratègia de salvar-se
darrera. Desafortunadament la mecànica del Genghis fa que molt poques vega
bé i, la majoria de les ocasions, s'acaba amb tot un costat del robot penjant (s
per al Genghis).
A tall de resum detallem en la següent taula quines són les longituds netes d
esquerra i dreta segons la pota que tinguem penjada. Els valors de la taula sen
al lloc on ha d'anar a parar la pota en fer el pas expressats en desplaçament
alfa. Primer considerem el cas en que o bé no hi ha potes penjades o les potes
del mateix extrem del robot:

potes penjades				long. esq.	long. dreta
0	1	4	5		
				$pas	$pas
SI				-$pas	0
	SI			0	-$pas
		SI		$pas	0
			SI	0	$pas
SI	SI			-$pas	-$pas
		SI	SI	$pas	$pas

I ara els casos en que hi ha potes del davant i potes del darrera en un precip
casos en que normalment l'estrarègia proposada no funciona:

potes penjades				long. esq.	long. dreta
0	1	4	5		
SI		SI		0	-$pas
SI			SI	-$pas	$pas
	SI	SI		$pas	-$pas
	SI		SI	-$pas	0
SI	SI	SI		-$pas	-$pas
SI	SI		SI	-$pas	-$pas
SI		SI	SI	0	-$pas
	SI	SI	SI	-$pas	0
SI	SI	SI	SI	-$pas	-$pas

Obstacles-esq

Entrades:

b-> Posició en curs del motor beta de la pota 0.

b-veina-> Posició en beta de la pota 1.

inclinacio-> Inclinació actual del robot.
estat-> Estat en beta de la pota 0.
estat-veina-> Estat en beta de la pota 1.
pas-> S'activa cada cop que la pota 0 fa un pas.
rebot-> S'activa cada cop que la pota 0 rebota contra un obstacle.

Sortides:
pas-esq-> Aportació a la longitud del pas de la banda esquerra que fa aques
pas-dret-> Aportació a la longitud del pas de la banda dreta.

Aquest comportament és el responsable de detectar que la pota 0 està sobre
que la pota 1 és en una posició més baixa. En aquest cas s'indueix una petita
cap a l'esquerra. Això evita que el robot s'escapi dels obstacles. Aquesta no é
estratègia òptima però es força útil per monitoritzar el comportament del Ge
posa en situacions cada cop més difícils).

Es consideren posicions vàlides de les potes 0 i 1 només aquelles que coincid
pota està recolzada en el terra, suportant pes del robot. Per la pota 0, a més,
estat fent rebots cosa que es considera indicació de que ha hagut de pujar un
per desambiguar algunes situacions, es llegeix l'actual inclinació del robot
aquest comportament si està aixecat del davant ja que de vegades, en baixar d
dona que la pota 0 està més alta que la pota 1 (per efectes dels beta balanços
estat fent rebots (per haver xocat contra el terra). Aquesta situció només es d
volem detectar per l'inclinació que en cada cas té el robot.

Obstacles-dreta
Entrades:
b-> Posició en curs del motor beta de la pota 1.
b-veina-> Posició en beta de la pota 0.
inclinacio-> Inclinació actual del robot.
estat-> Estat en beta de la pota 1.
estat-veina-> Estat en beta de la pota 0.
pas-> S'activa cada cop que la pota 1 fa un pas.
rebot-> S'activa cada cop que la pota 1 rebot en un obstacle.
Sortides:
pas-esq-> Aportació a la longitud del pas de la banda esquerra que fa aques
pas-dret-> Aportació a la longitud del pas de la banda dreta.

Comportament anàleg a l'anterior però dedicat a detectar la presència d'obst
robot.

ANNEX I:

LLISTATS DELS PROGRAMES