Planification d'actions concurrentes sous contraintes et incertitude

Éric Beaudry

Département d'informatique
Faculté des sciences
Université de Sherbrooke

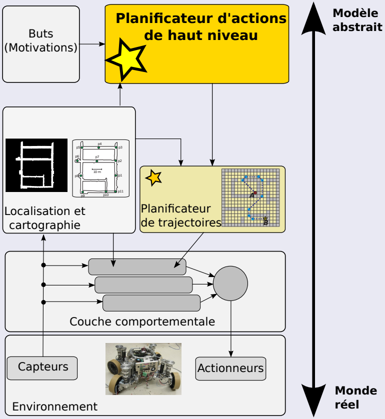
3 Mars 2011

http://planiart.usherbrooke.ca/~eric
mailto:eric.beaudry@usherbrooke.ca

UNIVERSITÉ DE SHERBROOKE

Hypothèses

Architecture décisionnelle (décomposition du problème)



Hypothèses typiques pour la planification de haut niveau

1. Domaine fini
2. Monde statique
3. **Déterministe**
4. **Plans séquentiels**
5. Effets instantanés
6. Observabilité totale

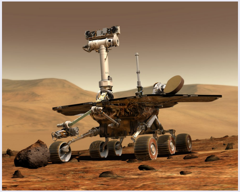
[Nau, Ghallab et Traverso 2004]

Motivation : Actions concurrentes + Incertitude

AAAI Robot Challenge 2005 et 2006 [Michaud etal. 2007]

- Localisation difficile dans une foule
- Incertitude de la durée des déplacements

Mars Exploration Rovers (MER) @ NASA [Bresina etal. 2002]



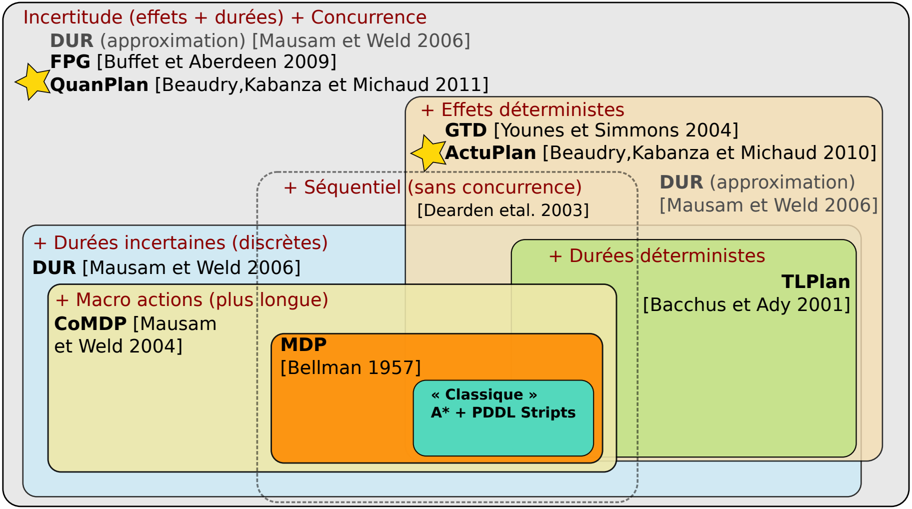
- Carte partiellement connue
- Incertitude de la durée des déplacements et des téléchargements
- Exécution d'actions concurrentes

Objectifs de recherche

Hypothèses retenues

- H1. ~~Domaine fini~~ $\Rightarrow$ **représentation hybride (temps continu)**
- H2. **Monde statique**
- H3. ~~Déterministe~~ $\Rightarrow$ **Probabiliste** (durée + effets)
- H4. ~~Plans séquentiels~~ $\Rightarrow$ **Actions concurrentes**
- H5. **Effets instantanés** (effets au début OU à la fin)
- H6. **Observabilité totale**

Classification des travaux connexes



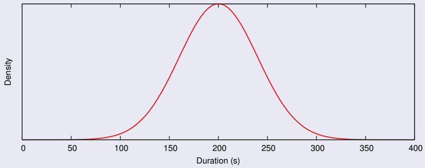
Concurrence + incertitude

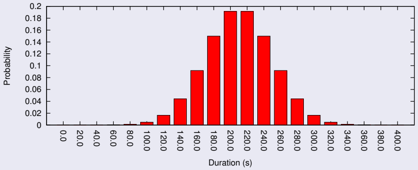
DUR : *Concurrent MDP* (CoMDP) [Mausam et Weld, 2008]

Un **état *interwoven*** est défini par $s = (X, \Delta)$ où :

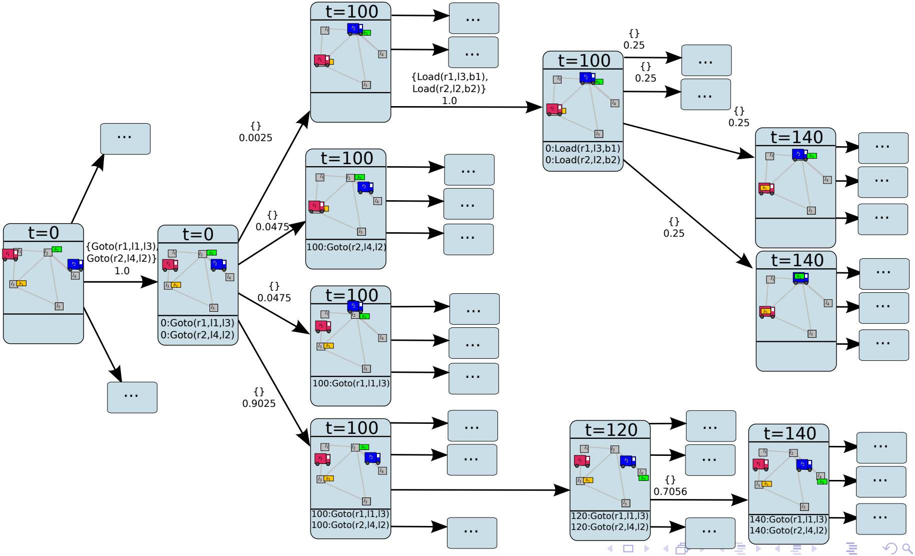
- X représente l'état présent
- $\Delta = \{(\delta_1, a_1), \ldots, (\delta_n, a_n)\}$ est les actions en exécution
- $\delta_i \in \mathbb{N}$ représente le temps écoulé depuis le début de l'action
- $\mathcal{A} = A^2 \cup a_{noop}$ est l'ensemble des actions combinées

Discrétisation du temps





Discrétisation ⇒ Espace d'états immense !



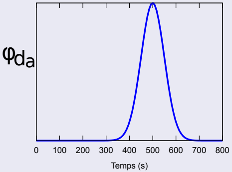
ACTUPLAN

- ACTUPLAN : *Action Concurrency Time Uncertainty Planner*
- Représentation compacte d'états
  - Variables discrètes X : partie non temporelle
  - Variables aléatoires continues : temps
  - *Généralisation aux variables numériques*
- Deux versions
  - ACTUPLAN^{nc} : Planificateur non conditionnel
  - ACTUPLAN : Planificateur conditionnel

Incertitude sur le temps

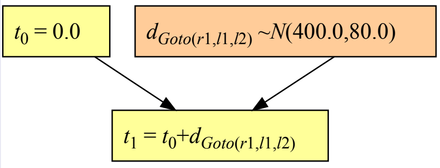
Durée des actions

- **Variables aléatoires continues** $D = \{d_a \mid a \in A\}$
- Fonctions de densité de probabilité ϕ_{d_a}
- Exemple $d_a \sim N(500, 50)$



Événements : **Début** et **fin** des actions

- **Variables aléatoires continues** $T = \{t_0, t_1, \ldots\}$
- Réseau bayésien : relations de dépendance



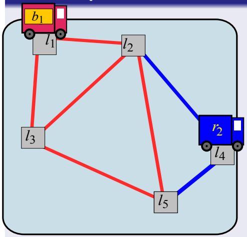
États

Définition

Un **état** s est défini par $s = (\mathcal{U}, \mathcal{V}, \mathcal{R})$

- $\mathcal{U} : X \to \cup_{x \in X} Dom(x)$ retourne l'**assignation courante**
- $\mathcal{V} : X \to T$ retourne le ***valid time***
- $\mathcal{R} : X \to T$ retourne le ***release time***

Exemple



s_0
$C_{r_1} = l_1 @ t_0 / t_0$
$C_{r_2} = l_4 @ t_0 / t_0$
$C_{b_1} = r_1 @ t_0 / t_0$

t_0
$= 0$

Légende :

$x = \mathcal{U}(x) @ \mathcal{V}(x) / \mathcal{R}(x)$

Application d'une action dans un état ($\Rightarrow s_1$)

Algorithme APPLY

```
function APPLY(s, a)
  s' ← s
  t_conds ← max_{x∈vars(conds(a))} s.V(x)
  t_release ← max_{x∈vars(effects(a))} s.R(x)
  t_start ← max(t_conds, t_release)
  t_end ← t_start + d_a
  for each c ∈ a.coverall
    for each x ∈ vars(c)
      s'.R(x) ← max(s'.R(x), t_end)
  for each e ∈ a.estart
    s'.U(e.x) ← eval(e.exp)
    s'.V(e.x) ← t_start
    s'.R(e.x) ← t_start
  for each e ∈ a.eend
    s'.U(e.x) ← eval(e.exp)
    s'.V(e.x) ← t_end
    s'.R(e.x) ← t_end
  returns s'
```

$Goto(r_1, l_1, l_2)$

cond. *at start* : $C_{r_1} = l_1$

eff. *at end* : $C_{r_1} \leftarrow l_2$

Exemple

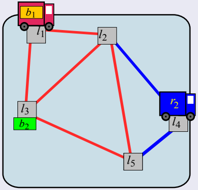
$Goto(r_1, l_1, l_2)$

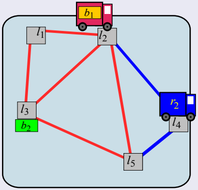
s_0

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_1	t_0	t_0
C_{r2}	l_4	t_0	t_0
C_{b1}	r_1	t_0	t_0
C_{b2}	l_3	t_0	t_0

s_1

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_1
C_{r2}	l_4	t_0	t_0
C_{b1}	r_1	t_0	t_0
C_{b2}	l_3	t_0	t_0





Temps début action : t_0

Temps fin action : $t_1 = t_0 + d_{Goto(r1,l1,l2)}$

Application d'une action dans un état ($\Rightarrow s_2$)

Algorithme Apply

function Apply(s, a)
  $s' \leftarrow s$
  $t_{conds} \leftarrow \max_{x \in vars(conds(a))} s.\mathcal{V}(x)$
  $t_{release} \leftarrow \max_{x \in vars(effects(a))} s.\mathcal{R}(x)$
  $t_{start} \leftarrow max(t_{conds}, t_{release})$
  $t_{end} \leftarrow t_{start} + d_a$
  for each $c \in a.coverall$
    for each $x \in vars(c)$
      $s'.\mathcal{R}(x) \leftarrow max(s'.\mathcal{R}(x), t_{end})$
  for each $e \in a.estart$
    $s'.\mathcal{U}(e.x) \leftarrow eval(e.exp)$
    $s'.\mathcal{V}(e.x) \leftarrow t_{start}$
    $s'.\mathcal{R}(e.x) \leftarrow t_{start}$
  for each $e \in a.eend$
    $s'.\mathcal{U}(e.x) \leftarrow eval(e.exp)$
    $s'.\mathcal{V}(e.x) \leftarrow t_{end}$
    $s'.\mathcal{R}(e.x) \leftarrow t_{end}$
  returns s'

$Goto(r_2, l_4, l_2)$

cond. *at start* : $C_{r_2} = l_4$

eff. *at end* : $C_{r_1} \leftarrow l_2$

Exemple

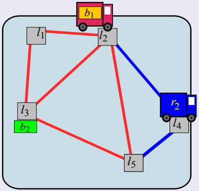
$Goto(r_2, l_4, l_2)$

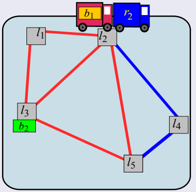
s_1

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_1
C_{r2}	l_4	t_0	t_0
C_{b1}	r_1	t_0	t_0
C_{b2}	l_3	t_0	t_0

s_2

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_1
C_{r2}	l_2	t_2	t_2
C_{b1}	r_1	t_0	t_0
C_{b2}	l_3	t_0	t_0





Temps début : t_0

Temps fin : $t_2 = t_0 + d_{Goto(r2,l4,l2)}$

Application d'une action dans un état ($\Rightarrow s_3$)

Algorithme APPLY

function APPLY(s, a)
  $s' \leftarrow s$
  $t_{conds} \leftarrow \max_{x \in vars(conds(a))} s.\mathcal{V}(x)$
  $t_{release} \leftarrow \max_{x \in vars(effects(a))} s.\mathcal{R}(x)$
  $t_{start} \leftarrow max(t_{conds}, t_{release})$
  $t_{end} \leftarrow t_{start} + d_a$
  for each $c \in a.coverall$
    for each $x \in vars(c)$
      $s'.\mathcal{R}(x) \leftarrow max(s'.\mathcal{R}(x), t_{end})$
  for each $e \in a.estart$
    $s'.\mathcal{U}(e.x) \leftarrow eval(e.exp)$
    $s'.\mathcal{V}(e.x) \leftarrow t_{start}$
    $s'.\mathcal{R}(e.x) \leftarrow t_{start}$
  for each $e \in a.eend$
    $s'.\mathcal{U}(e.x) \leftarrow eval(e.exp)$
    $s'.\mathcal{V}(e.x) \leftarrow t_{end}$
    $s'.\mathcal{R}(e.x) \leftarrow t_{end}$
  returns s'

$Unload(r_1, l_2, b_1)$

cond. *at start* : $C_{b_1} = r_1$

cond. *over all* : $C_{r_1} = l_2$

eff. *at end* : $C_{b_1} \leftarrow l_2$

Exemple

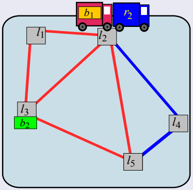
$s_2 \xrightarrow{Unload(r_1, l_2, b_1)} s_3$

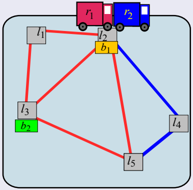
s_2

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_1
C_{r2}	l_2	t_2	t_2
C_{b1}	r_1	t_0	t_0
C_{b2}	l_3	t_0	t_0

s_3

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_3
C_{r2}	l_2	t_2	t_2
C_{b1}	l_2	t_3	t_3
C_{b2}	l_3	t_0	t_0





Temps début : $max(t_0, t_1) = t_1$

Temps fin : $t_3 = t_1 + d_{Unload(r1,l2,b1)}$

ActuPlan^{nc} : planificateur non conditionnel

- Algorithme **recherche meilleur en premier** (*Best-First-Search*) [1]
- But : trouver état s tel que $P(s \models \mathcal{G}) \geq \alpha$
- Fonctions d'évaluation $f = cost(s)$
  - *Makespan* : $cost(s) = E\left[\max_{x \in X} s.\mathcal{V}(x)\right]$
  - Somme du coût des actions
  - Combinaison linéaire
- Estimation du coût final à l'aide d'une heuristique **admissible**
- Sortie : plan d'actions partiellement ordonnées $\pi = \langle \{a_1, \ldots, a_n\}, \{a_i \prec a_j, \ldots, a_k \prec a_l\} \rangle$

1. À ne pas confondre avec ***Greedy*** *Best-First-Search*

Exemple de recherche

Sous-ensemble de l'espace d'états

s_0 —*Goto*(r_1,l_1,l_2)→ s_1 —*Goto*(r_2,l_4,l_2)→ s_2 —*Unload*(r_1,l_2,b_1)→ s_3 —*Load*(r_2,l_2,b_1)→ s_4 —*Goto*(r_2,l_2,l_4)→ s_5 —*Unload*(r_2,l_4,b_1)→ s_6

s_0

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_1	t_0	t_0
C_{r2}	l_4	t_0	t_0
C_{b1}	r_1	t_0	t_0
C_{b2}	l_3	t_0	t_0

s_1

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_1
C_{r2}	l_4	t_0	t_0
C_{b1}	r_1	t_0	t_0
C_{b2}	l_3	t_0	t_0

s_2

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_1
C_{r2}	l_4	t_2	t_2
C_{b1}	r_1	t_0	t_0
C_{b2}	l_3	t_0	t_0

s_3

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_3
C_{r2}	l_2	t_2	t_2
C_{b1}	l_2	t_3	t_3
C_{b2}	l_3	t_0	t_0

s_4

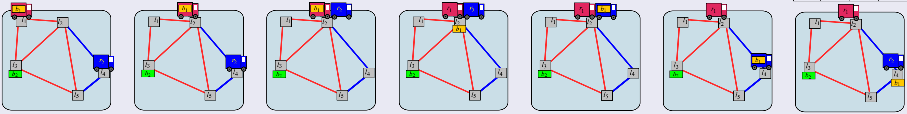
x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_3
C_{r2}	l_2	t_2	t_5
C_{b1}	r_2	t_5	t_5
C_{b2}	l_3	t_0	t_0

s_5

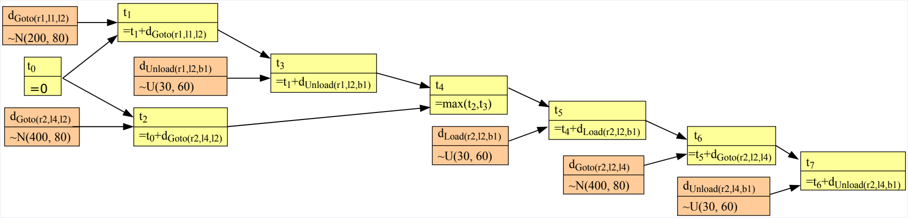
x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_3
C_{r2}	l_4	t_6	t_6
C_{b1}	r_2	t_5	t_5
C_{b2}	l_3	t_0	t_0

s_6

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_3
C_{r2}	l_4	t_6	t_7
C_{b1}	l_4	t_7	t_7
C_{b2}	l_3	t_0	t_0



Sous-ensemble du réseau bayésien



Plan

Chemin trouvé dans l'espace d'états

s_0

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_1	t_0	t_0
C_{r2}	l_4	t_0	t_0
C_{b1}	r_1	t_0	t_0
C_{b2}	l_3	t_0	t_0

$\xrightarrow{Goto(r_1,l_1,l_2)}$ s_1

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_1
C_{r2}	l_4	t_0	t_0
C_{b1}	r_1	t_0	t_0
C_{b2}	l_3	t_0	t_0

$\xrightarrow{Goto(r_2,l_4,l_2)}$ s_2

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_1
C_{r2}	l_4	t_2	t_2
C_{b1}	r_1	t_0	t_0
C_{b2}	l_3	t_0	t_0

$\xrightarrow{Unload(r_1,l_2,b_1)}$ s_3

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_3
C_{r2}	l_2	t_2	t_2
C_{b1}	l_2	t_3	t_3
C_{b2}	l_3	t_0	t_0

$\xrightarrow{Load(r_2,l_2,b_1)}$ s_4

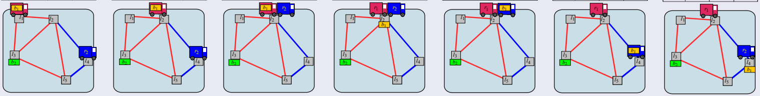
x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_3
C_{r2}	l_2	t_2	t_5
C_{b1}	r_2	t_5	t_5
C_{b2}	l_3	t_0	t_0

$\xrightarrow{Goto(r_2,l_2,l_4)}$ s_5

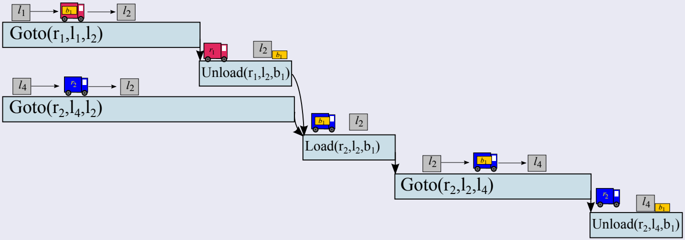
x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_3
C_{r2}	l_4	t_6	t_6
C_{b1}	r_2	t_5	t_5
C_{b2}	l_3	t_0	t_0

$\xrightarrow{Unload(r_2,l_4,b_1)}$ s_6

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_3
C_{r2}	l_4	t_6	t_7
C_{b1}	l_4	t_7	t_7
C_{b2}	l_3	t_0	t_0



Plan extrait



Évaluation du réseau bayesien

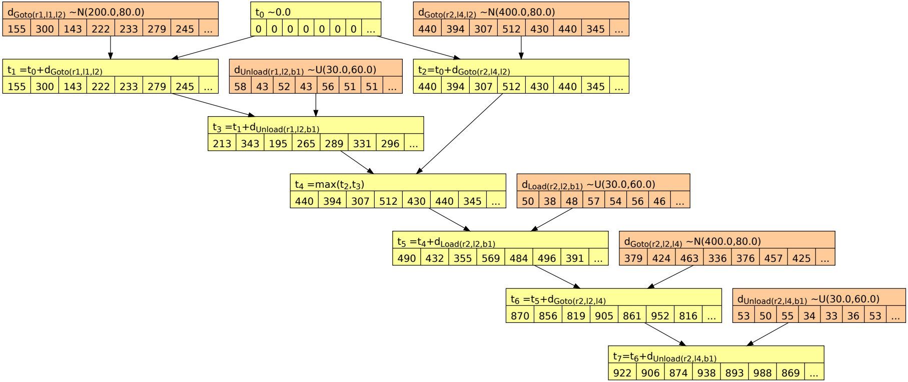
Requêtes d'évaluation

- $P(s \models \mathcal{G})$
- Évaluation de la fonction heuristique

Algorithme d'inférence

- Évaluation exacte (solution analytique) impossible en pratique
  - Si $t_1 \sim N(\mu_1, \sigma_1)$ et $t_2 \sim N(\mu_2, \sigma_2) \Rightarrow max(t_1, t_2)$ ne suit pas une loi normale
  - Variété de lois de distribution
- Estimation par échantillonnage direct (solution approximative)
- Estimation incrémentale

Estimation par échantillonnage



Estimation

$Goto(r_1,l_1,l_2)$ → $Goto(r_2,l_4,l_2)$ → $Unload(r_1,l_2,b_1)$ → $Load(r_2,l_2,b_1)$ → $Goto(r_2,l_2,l_4)$ → $Unload(r_2,l_4,b_1)$

s_0

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_1	t_0	t_0
C_{r2}	l_4	t_0	t_0
C_{b1}	r_1	t_0	t_0
C_{b2}	l_3	t_0	t_0

s_1

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_1
C_{r2}	l_4	t_0	t_0
C_{b1}	r_1	t_0	t_0
C_{b2}	l_3	t_0	t_0

s_2

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_1
C_{r2}	l_4	t_2	t_2
C_{b1}	r_1	t_0	t_0
C_{b2}	l_3	t_0	t_0

s_3

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_3
C_{r2}	l_2	t_2	t_2
C_{b1}	l_2	t_3	t_3
C_{b2}	l_3	t_0	t_0

s_4

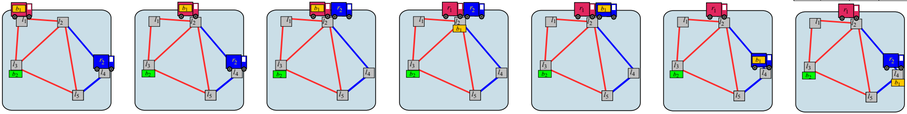
x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_3
C_{r2}	l_2	t_2	t_5
C_{b1}	r_2	t_5	t_5
C_{b2}	l_3	t_0	t_0

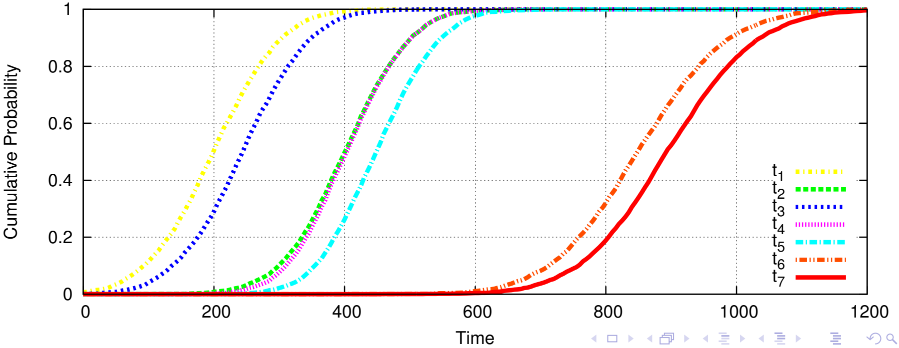
s_5

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_3
C_{r2}	l_4	t_6	t_6
C_{b1}	r_2	t_5	t_5
C_{b2}	l_3	t_0	t_0

s_6

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_2	t_1	t_3
C_{r2}	l_4	t_6	t_7
C_{b1}	l_4	t_7	t_7
C_{b2}	l_3	t_0	t_0

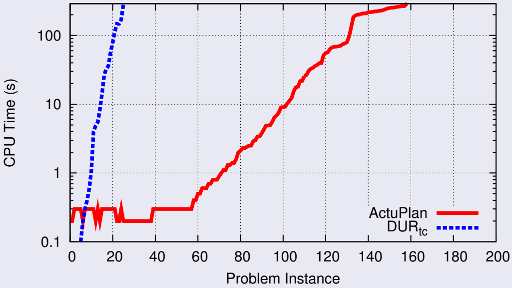




Résultats ACTUPLAN^{nc}

Domaine Transport (*International Planning Competition*)

- 192 Tests aléatoires (4 à 12 lieux, 1 à 4 camions, 1 à 8 buts)
- Tous les tests ont une solution
- Temps alloué : 300 secondes ; Mémoire disponible : 2Go

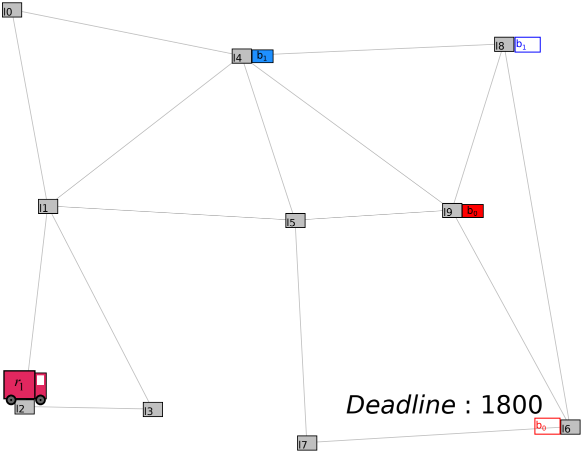


ACTUPLAN : planificateur conditionnel

Deux passes

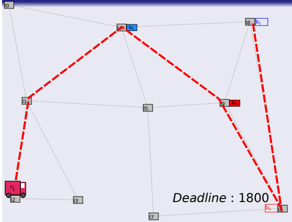
1. Génération de plans non conditionnels $\{\pi \mid P(\pi) \geq \beta\}$
2. Fusion des plans pour obtenir un plan conditionnel $\pi \mid P(\pi) \geq \alpha \geq \beta$

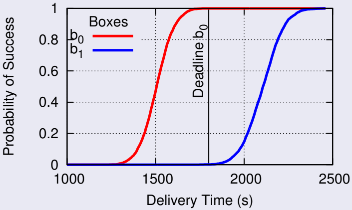
Problème



Deux plans non conditionnels

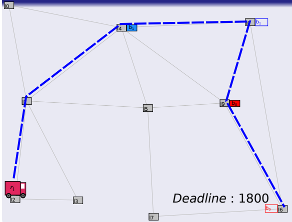
Plan π_a

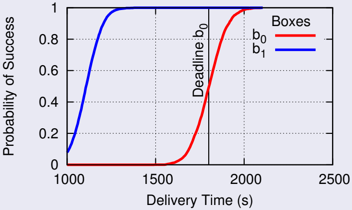




$$Cost(\pi_a) = 2105$$
$$P(\pi_a) = 1.0$$

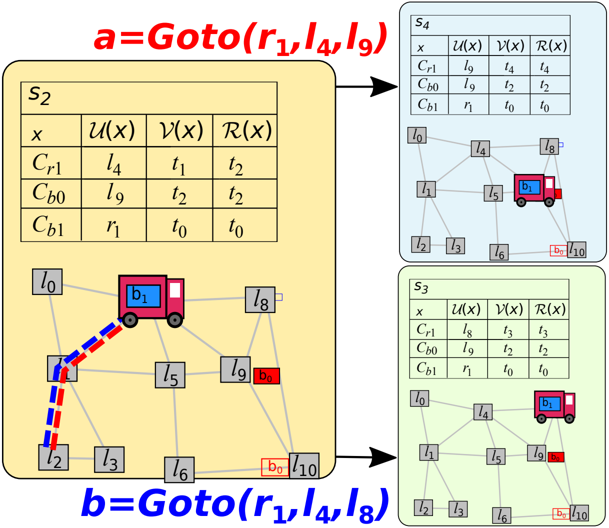
Plan π_b





$$Cost(\pi_b) = 1800$$
$$P(\pi_b) = 0.5$$

Retarder la décision



s_2

x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_4	t_1	t_2
C_{b0}	l_9	t_2	t_2
C_{b1}	r_1	t_0	t_0

s_4

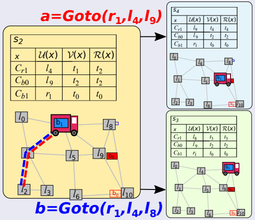
x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_9	t_4	t_4
C_{b0}	l_9	t_2	t_2
C_{b1}	r_1	t_0	t_0

s_3

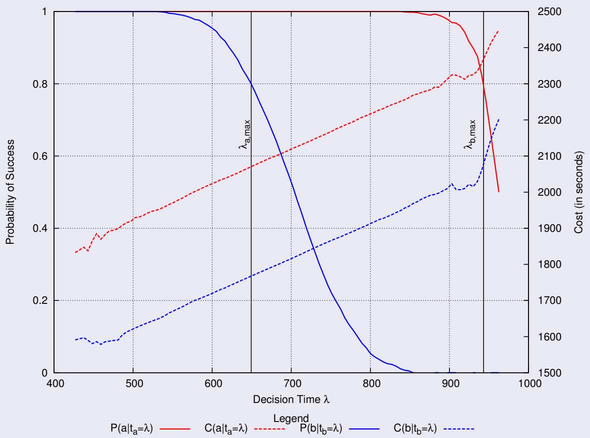
x	$\mathcal{U}(x)$	$\mathcal{V}(x)$	$\mathcal{R}(x)$
C_{r1}	l_8	t_3	t_3
C_{b0}	l_9	t_2	t_2
C_{b1}	r_1	t_0	t_0

Temps critiques $\lambda_{a,max}$ et $\lambda_{b,max}$

Le choix

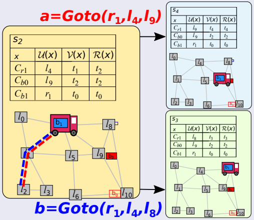


Temps maximums

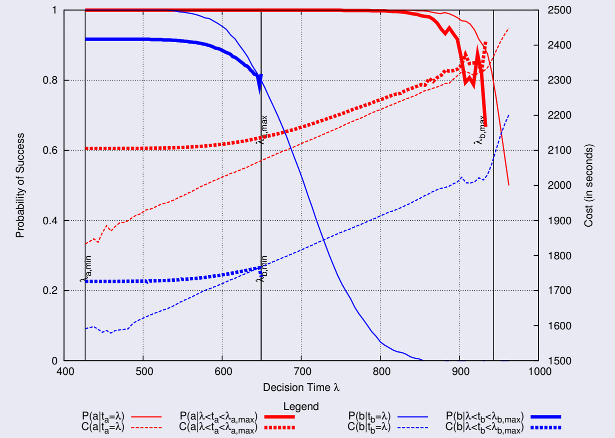


Temps critiques $\lambda_{a,min}$ et $\lambda_{b,min}$

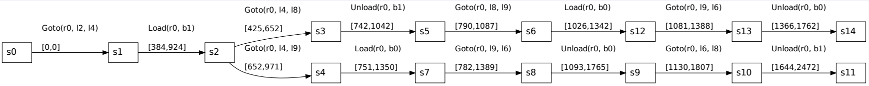
Le choix



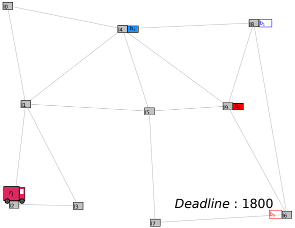
La décision



Plan conditionnel généré



$$Cost(\pi) = 2013$$
$$P(\pi) = 0.974$$



Résultats ActuPlan

Domaine Transport (*International Planning Competition*)

Size		ActuPlan^{nc}			ActuPlan			DUR$_{tc}$		
$\|R\|$	$\|\mathcal{G}\|$	CPU	*P*	*Cost*	CPU	*P*	*Cost*	CPU	*P*	*Cost*
1	2	0.3	1.00	2105.5	0.3	0.96	1994.5	44.9	0.99	1992.4
1	3	0.3	1.00	2157.3	0.6	0.93	1937.9	214.5	0.00	-
1	4	0.3	0.98	2428.0	0.5	0.98	2426.2	192.6	0.00	-
2	2	0.2	1.00	1174.0	0.4	1.00	1179.2	302.3	0.00	-
2	3	0.3	1.00	1798.3	1.8	0.93	1615.2	288.3	0.00	-
2	4	0.3	1.00	1800.0	4.2	0.91	1500.5	282.8	0.00	-
3	6	4.1	0.97	1460.1	300	0.97	1464.5	307.9	0.00	-

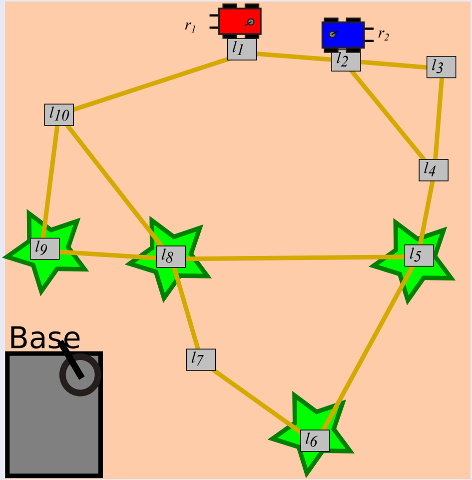
QUANPLAN

- QUANPLAN : *Quantum State Space Planner*
- Incertitude sur les **effets** et la **durée** des actions
- Métrique unifiée
  - ~~Seuil sur la probabilité de succès α~~
  - Coût = Coût des actions + *Makespan* + Pénalité d'échec
  - $Cost(\pi) = \sum_{a \in A_\pi} C(a) + \alpha E\left[\max_{x \in X} s.\mathcal{V}(x)\right] + P(s \not\models \mathcal{G}) \times Penality$

    où $s = Apply^*(s_0, \pi)$
- Défis
  1. Modéliser des états avec actions incertaines en cours d'exécutions
  2. Choisir des actions alors les effets de d'autres actions sont encore indéterminées

Exemple : Mars *Rovers*

Carte à explorer



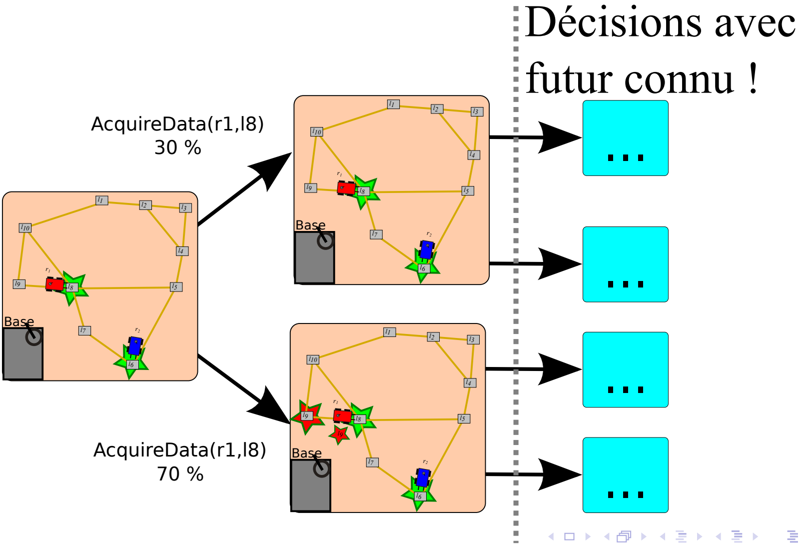
But

Analyser les sites l_5, l_6, l_8 et l_9 à l'aide d'un capteur spécialisé et transmettre les résultats à la base.

Actions

- Goto
- InitSensor
- AcquireData
- Transmit

Approche naïve



Superposition d'états (états quantiques)

État partiel

Un **état partiel déterminé** est défini par $\breve{s}_{\breve{X}} = (\breve{\mathcal{U}}, \breve{\mathcal{V}}, \breve{\mathcal{R}})$ où $\breve{X} \subseteq X$.

Superposition d'états partiels

- Une **superposition d'états partiels déterminés** $\overline{s}_{\breve{X}}$ est un ensemble d'états partiels déterminé où chaque $\breve{s}_{\breve{X}} \in \overline{s}_{\breve{X}}$ est défini sur le même ensemble $\breve{X} \subseteq X$.
- Probabilité d'observation : $P(\breve{s}_{\breve{X}} \mid \overline{s}_{\breve{X}})$.

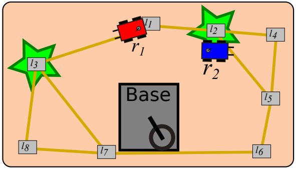
État quantique

Un **état quantique** q est défini par

$$q = \left\{ \overline{s}_{\breve{X}_1}, \ldots, \overline{s}_{\breve{X}_n} \mid \breve{X}_1, \ldots, \breve{X}_n \text{ est une partition de} X \right\}.$$

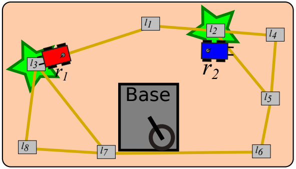
États complètements déterminés (ActuPlan)

s_1			
x	U(x)	V(x)	R(x)
C_{r1}	l_1	t_0	t_0
C_{r2}	l_2	t_0	t_0
I_{r1}	T	t_1	t_1
I_{r2}	T	t_0	t_0
$D_{r1,l3}$	F	t_0	t_0
$D_{r2,l2}$	F	t_0	t_0
B_{l2}	F	t_0	t_0
...	...	...	...



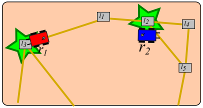
Goto(r_1,l_1,l_3)

s_2			
x	U(x)	V(x)	R(x)
C_{r1}	l_3	t_2	t_2
C_{r2}	l_2	t_0	t_0
I_{r1}	T	t_1	t_1
I_{r2}	T	t_0	t_0
$D_{r1,l3}$	F	t_0	t_0
$D_{r2,l2}$	F	t_0	t_0
B_{l2}	F	t_0	t_0
...	...	...	...



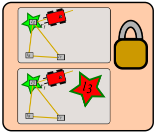
Génération d'états quantiques

$q_2 = s_2$			
x	U(x)	V(x)	R(x)
C_{r1}	l_3	t_2	t_2
C_{r2}	l_2	t_0	t_0
I_{r1}	T	t_1	t_1
I_{r2}	T	t_0	t_0
$D_{r1,l3}$	F	t_0	t_0
$D_{r2,l2}$	F	t_0	t_0
B_{l2}	F	t_0	t_0
...	...	...	...



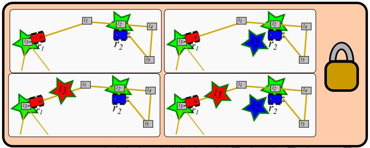
Acquire
(r_1, l_3)
1.0

q_3		
$s_{3,0}$		
P	1.0	
C_{r1} @t_2/t_4	l_3	
C_{r2} @t_0/t_0	l_2	
I_{r2} @t_0/t_0	T	
$D_{r2,l2}$ @t_0/t_0	F	
...	...	
$\overline{s}_{3,1}$		
P	0.3	0.7
I_{r1} @t_4/t_4	F	F
$D_{r1,l3}$ @t_4/t_4	F	T



Acquire
(r_2, l_2)
1.0

q_4		
$s_{4,0}$		
P	1.0	
C_{r1} @t_2/t_4	l_3	
C_{r2} @t_0/t_5	l_2	
...	...	
$\overline{s}_{4,1}$		
P	0.3	0.7
I_{r1} @t_4/t_4	F	F
$D_{r1,l3}$ @t_4/t_4	F	T
$\overline{s}_{4,2}$		
P	0.3	0.7
I_{r2} @t_5/t_5	F	F
$D_{r2,l2}$ @t_5/t_5	F	T

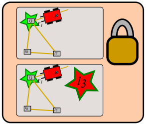


Détermination retardée à l'observation

Algorithme

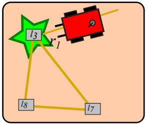
```
function OBSERVE(x ∈ X, q = {s̄_X̌1, ..., s̄_X̌n})
  i ← find i such that x ∈ X̌_i
  Q_r ← ∅
  for each š_X̌i ∈ s̄_X̌i
    q' ← q \ s̄_X̌i
    t_determ ← max_{x'∈X̌i} š_X̌i.V(x')
    š'_X̌i ← š_X̌i
    s̄'_X̌i ← {š'_X̌i}
    P(š'_X̌i | s̄'_X̌i) ← 1.0
    q' ← q' ∪ s̄'_X̌i
    for each X̌ ∈ q'
      for each x' ∈ X̌
        š'_X̌i.V(x') ← max(š'_X̌i.V(x'), t_determ)
    P(q' | ActionObserve(x), q) ← P(š_X̌i | s̄_X̌i)
    Q_r ← Q_r ∪ q'
  return Q_r
```

q_3		
$s_{3,0}$		
P	1.0	
C_{r1} @t_2/t_4	l_3	
C_{r2} @t_0/t_0	l_2	
I_{r2} @t_0/t_0	T	
$D_{r2,l2}$ @t_0/t_0	F	
...	...	
$\bar{s}_{3,1}$		
P	0.3	0.7
I_{r1} @t_4/t_4	F	F
$D_{r1,l3}$ @t_4/t_4	F	T



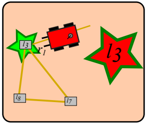
Observe($D_{r1,l3}$) 0.3

q_4	
$s_{4,0}$	
P	1.0
C_{r1} @t_4/t_4	l_3
C_{r2} @t_4/t_4	l_2
I_{r1} @t_4/t_4	F
I_{r2} @t_4/t_4	T
$D_{r1,l3}$ @t_4/t_4	F
$D_{r2,l2}$ @t_4/t_4	F
...	...



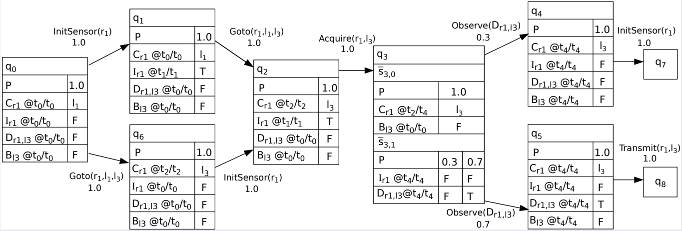
Observe($D_{r1,l3}$) 0.7

q_5	
$s_{5,0}$	
P	1.0
C_{r1} @t_4/t_4	l_3
C_{r2} @t_4/t_4	l_2
I_{r1} @t_4/t_4	F
I_{r2} @t_4/t_4	T
$D_{r1,l3}$ @t_4/t_4	T
$D_{r2,l2}$ @t_4/t_4	F
...	...

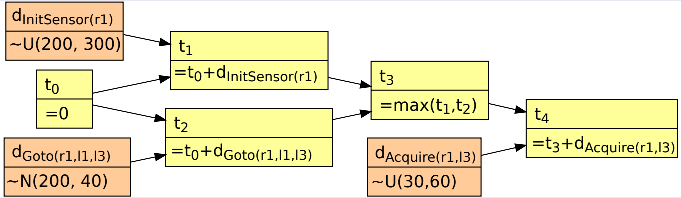


Espace d'états quantiques

Sous espace d'états



Réseau bayésien partiel



Génération de politiques simples (1)

MDP

Un problème est défini par $(Q, \mathcal{A}, C, P, s_0, \mathcal{G})$ où :

- Q set of quantum states (finite if horizon time is bounded)
- $\mathcal{A} = A \cup \{ActionObserve(x) \mid x \in X\}$ augmented set of actions
- $C = \{C_a \mid a \in A\}$ random variables costs of actions
- $P(q' \mid a, q)$ is the probabilistic transition model
- $q_0 \in Q$ initial quantum state (generally determined)
- $\mathcal{G}$ is the goal to satisfy

Génération de politiques simples (2)

Valeurs des états

$$J(q) = \begin{cases} \alpha E\left[\max_{x\in X} q.\mathcal{V}(x)\right] & \text{si } q \models \mathcal{G} \\ \min_{a\in A}\left(E[c_a] + \sum_{q'} P(q'|a,q)J(q')\right) & \text{sinon} \end{cases}$$

où α est le poids accordé au *Makespan*.

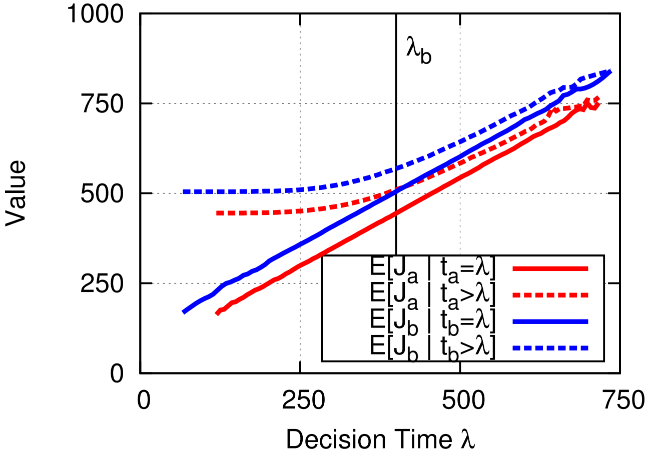
Extraction de la politique

$\pi(q) = \arg\min_{a\in\mathcal{A}} \left(E[c_a] + \sum_{q'} P(q'|a,q)J(q')\right)$

Algorithme

Labeled Real-Time Dynamic Programming (LRTDP) [Bonet et Geffner 2003]

Politiques avancées : conditions temporelles

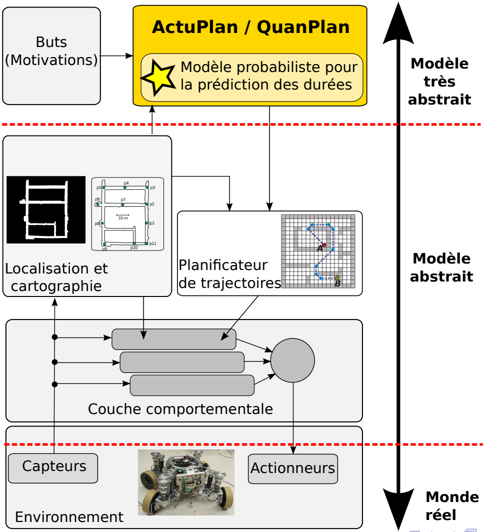


Résultats QuanPlan

Domaine des *Rovers*

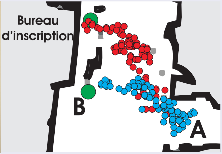
Problem		QuanPlan		DUR_{tc}		FPG	
$\|R\|$	$\|\mathcal{G}\|$	CPU	Cost	CPU	Cost	CPU	Cost
1	2	0.51	1927	4.99	1932	4	2340
1	3	1.33	2553	112	2563	>300	–
1	4	4.04	3001	>300	–	>300	–
1	5	53.3	3642	>300	–	>300	–
2	2	4.1	1305	65.2	1313	>300	–
2	3	14.9	2129	>300	–	>300	–
2	4	56.0	1285	>300	–	>300	–
2	5	>300	–	>300	–	>300	–

Intégration de d'ACTUPLAN et de QUANPLAN



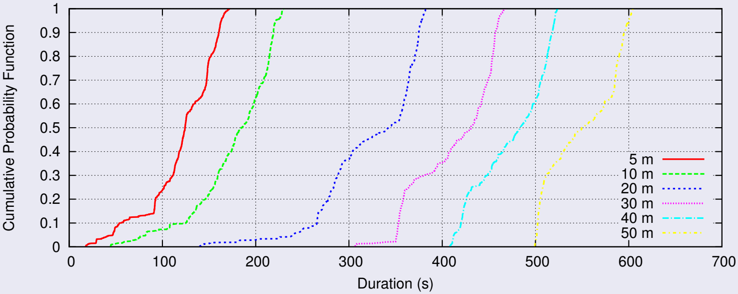
Modèle probabiliste durée déplacements (1)

Trace de Spartacus au *AAAI Robot Challenge* 2005



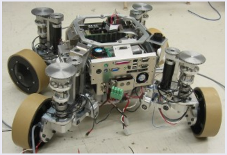
Distance effective : 128 m
Durée : 1049 s
Vitesse moyenne effective : 0.122 m/s
Durée trajet 20 m : $\mu = 322$ s, $\sigma = 52.5$ s

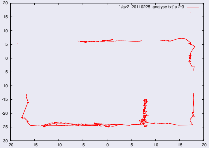
Modèle extrait



Modèle probabiliste durée déplacements (2)

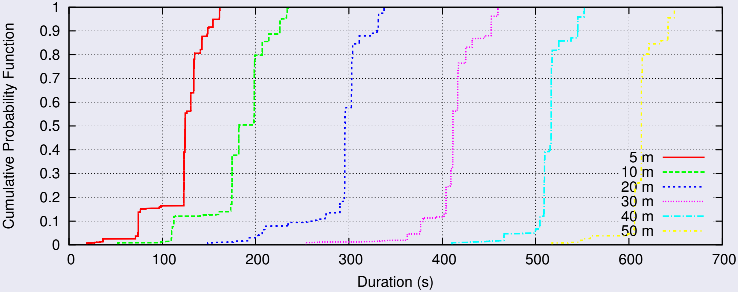
Trace d'AZIMUT-2 (2011-02-25)





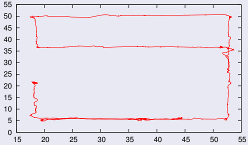
Distance effective : 187 m
Durée : 1778 s
Vitesse moyenne effective : 0.105 m/s
Durée trajet 20 m : $\mu = 292$ s, $\sigma = 33$ s

Modèle extrait



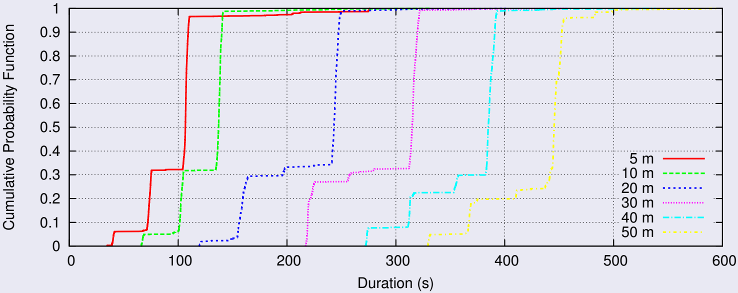
Modèle probabiliste durée déplacements (3)

Spartacus seul dans les corridors (2006-02-25)



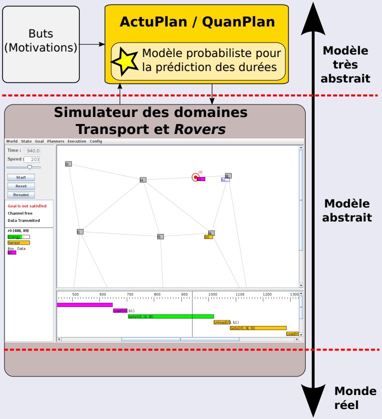
Distance effective : 270 m
Durée : 2256 s
Vitesse moyenne effective : 0.105 m/s
Durée trajet 20 m : $\mu = 217$ s, $\sigma = 23$ s

Modèle extrait



Validation

Architecture de validation



Caractéristiques du simulateur

- Génération de durées aléatoires selon des de lois de distribution (incluant modèle de déplacement créés à partir de traces réelles)
- Validation des plans d'ACTUPLAN

Contributions

Publié
Soumis et en révision
Sera soumis

Chapitre 1 : ActuPlan (JAIR)

- Variables aléatoires continues
- Réseaux bayésiens
- Plans non conditionnels quasi optimaux
- Plans conditionels quasi optimaux (borne inférieure calculable)

ICAPS 2010

ECAI 2010

Chapitre 2 : QuanPlan (AIJ)

- Généralisation incertitude effets
- États « quantiques »
- MDP / LTRDP
- Plans quasi optimaux

AAAI 2011

Chapitre 3 : Jeu IEEE CIG 2010

Chapitre 4 : Planification de Trajectoire (IROS 2010)

Contenu présenté

Contenu complémentaire

Journaux : *Journal of Artificial Intelligence Research (JAIR), Artificial Intelligence (AIJ).* Conférences : *International Conference on Automated Plannnig and Scheduling (ICAPS), Conference of the Association for the Advance in Artificial Intelligence(AAAI), European Conference on Artificial Intelligence (ECAI), IEEE Computational Intelligence and Games (CIG), IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).*

Travaux en cours et futurs

À terminer

1. Intégration ACTUPLAN et QUANPLAN architecture robotique
2. Compléter l'implémentation d'un parseur style PDDL
3. Expérimentation avec d'autres domaines d'IPC
4. Publication code source

À venir ... dans d'éventuels laboratoires d'accueil !

1. Approche *plan space*
2. Échantillonnage par importance (*importance sampling*)
3. Satisfaction partielle de buts (*Over-Subscription Planning*)
4. Stabilité des plans
5. Observabilité partielle

Fin de la présentation

Questions ?

Les travaux reliés à cette thèse ont été rendus possibles grâce à l'appui financier du Conseil de recherches en sciences naturelles et génie du Canada et du Fonds québécois de recherche sur la nature et les technologies.

CRSNG NSERC

Fonds de recherche sur la nature et les technologies Québec