

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Conclusion

Un cadre algébrique général pour représenter et résoudre des problèmes de décision séquentielle avec incertitudes, faisabilités et utilités

Cédric Pralet

Directeurs de thèse: Thomas Schiex et Gérard Verfaillie

7 Novembre 2007



Contexte de la thèse: **prise de décision automatique** (planification, ordonnancement, diagnostic...)

But: faire évoluer le système

- vers des états ou des suites d'états **préférés**,
- dans un contexte où l'état du monde peut être partiellement connu et où l'effet des actions peut être non déterministe (**incertitudes**)
- en tenant compte de certaines **contraintes sur les décisions**

Problèmes plus ou moins complexes suivant les aspects pris en compte

De nombreux cadres développés, de nombreuses extensions de ces cadres pour prendre en compte tel ou tel aspect

ex: CSP, réseaux bayésiens, MDP, formules booléennes quantifiées...

Objectif

Construire un **cadre algébrique générique** pour représenter **et** résoudre de tels problèmes

Intérêts:

- apporter une vision algébrique générique pour mieux comprendre les **liens entre différents cadres**
- apporter un cadre de modélisation **flexible**
- identifier des **propriétés** permettant d'appliquer tel ou tel algorithme ou d'obtenir telle ou telle complexité,
- définir des **algorithmes génériques** permettant une **fertilisation croisée** des développements entre cadres couverts et d'**éviter des travaux dupliqués**

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Conclusion

- 1 Exemple introductif
- 2 Le cadre PFU
- 3 Algorithmes génériques sur le cadre PFU

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Conclusion

- 1 Exemple introductif
- 2 Le cadre PFU
- 3 Algorithmes génériques sur le cadre PFU

Exemple introductif

Introduction

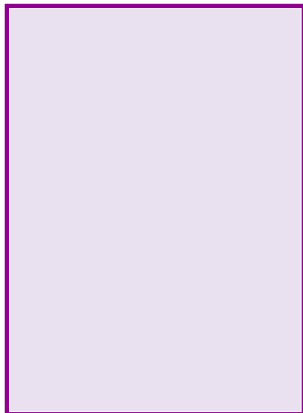
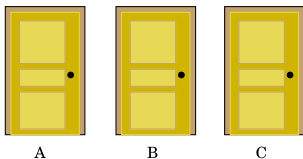
Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Conclusion

Jean doit décider d'ouvrir une porte
Un trésor derrière une des portes
Un voleur derrière une des portes
Jean et son compagnon Pierre peuvent
décider d'écouter à une des portes pour
tenter d'entendre le voleur



Exemple introductif

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

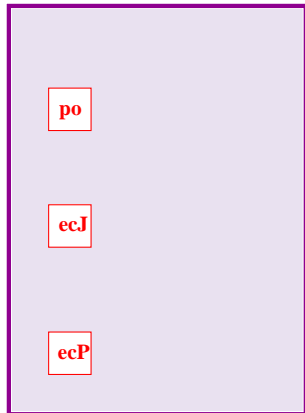
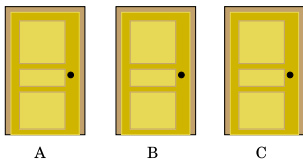
Conclusion

Jean doit décider d'ouvrir une porte

Un trésor derrière une des portes

Un voleur derrière une des portes

*Jean et son compagnon Pierre peuvent
décider d'écouter à une des portes pour
tenter d'entendre le voleur*



Variables de décision
(contrôlables)

Exemple introductif

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

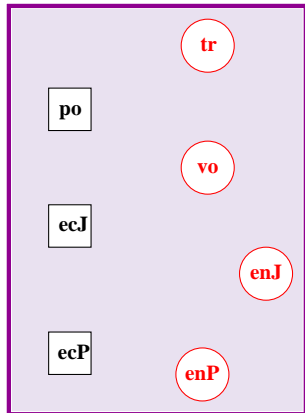
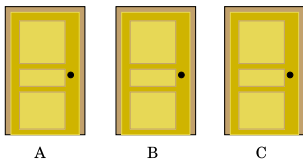
Conclusion

Jean doit décider d'ouvrir une porte

Un trésor derrière une des portes

Un voleur derrière une des portes

*Jean et son compagnon Pierre peuvent
décider d'écouter à une des portes **pour**
tenter d'entendre le voleur*



Variables de décision
(contrôlables)

Variables d'environnement
(incontrôlables)

Exemple introductif

Introduction

Exemple

Le cadre PFU

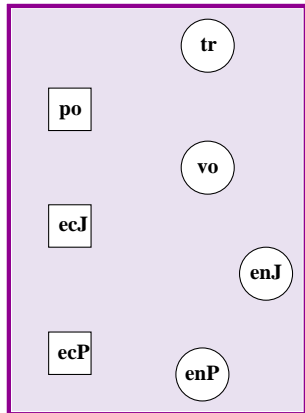
Algorithmes
génériques sur
le cadre PFU

Conclusion

Incertitudes sur l'état de
l'environnement et sur l'effet des actions:

*Le trésor et le voleur ne sont pas derrière
la même porte et toutes les situations
possibles sont équiprobables.*

*Probabilité d'entendre quelque chose:
fonction de la position du voleur.*



Exemple introductif

Introduction

Exemple

Le cadre PFU

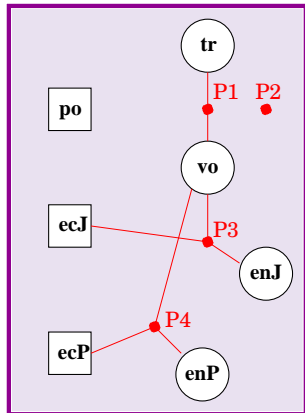
Algorithmes
génériques sur
le cadre PFU

Conclusion

Incertitudes sur l'état de
l'environnement et sur l'effet des actions:

*Le trésor et le voleur ne sont pas derrière
la même porte et toutes les situations
possibles sont équiprobables.*

*Probabilité d'entendre quelque chose:
fonction de la position du voleur.*



**Fonctions locales de
plausibilité**

$P_1 : vo \neq tr, P_2 : 1/6$

$P_3 : P(en_J | ec_J, vo)$

Exemple introductif

Introduction

Exemple

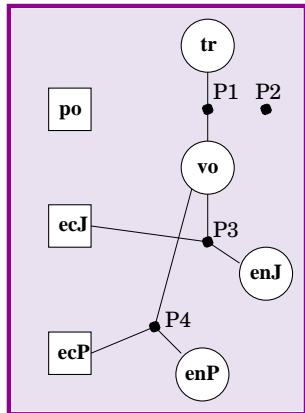
Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Conclusion

Contraintes sur les décisions:

Jean et Pierre ne peuvent pas écouter à la même porte et ouvrir la porte A est interdit.



Exemple introductif

Introduction

Exemple

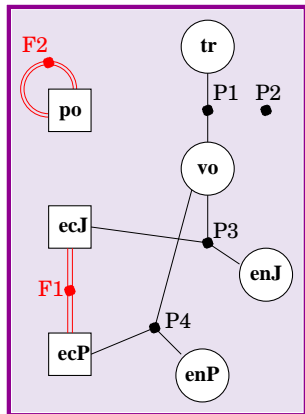
Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Conclusion

Contraintes sur les décisions:

Jean et Pierre ne peuvent pas écouter à la même porte et ouvrir la porte A est interdit.



Fonctions locales de faisabilité

$$F_1 : ec_J \neq ec_P$$

$$F_2 : po \neq A$$

Exemple introductif

Introduction

Exemple

Le cadre PFU

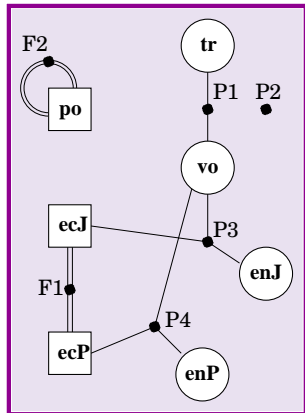
Algorithmes
génériques sur
le cadre PFU

Conclusion

Préférences:

Trésor: +10,000€

Voleur: -4,000€



Exemple introductif

Introduction

Exemple

Le cadre PFU

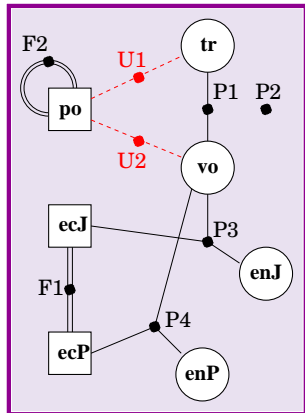
Algorithmes
génériques sur
le cadre PFU

Conclusion

Préférences:

Trésor: +10,000€

Voleur: -4,000€



**Fonctions locales
d'utilité**

$U_1 : po = tr$ (+10 K€)

$U_2 : po = vo$ (-4 K€)

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Conclusion

1^{ère} notion importante du cadre:

Connaissance exprimée sous forme de variables
et de fonctions locales portant sur ces variables

Aspect “composite”

Requête sur le problème

Règles de décision **maximisant l'utilité espérée** si d'abord Pierre et Jean écoutent chacun à une porte, et ensuite Jean choisit une porte à ouvrir en fonction de ce qui a été entendu?

Introduction

Exemple

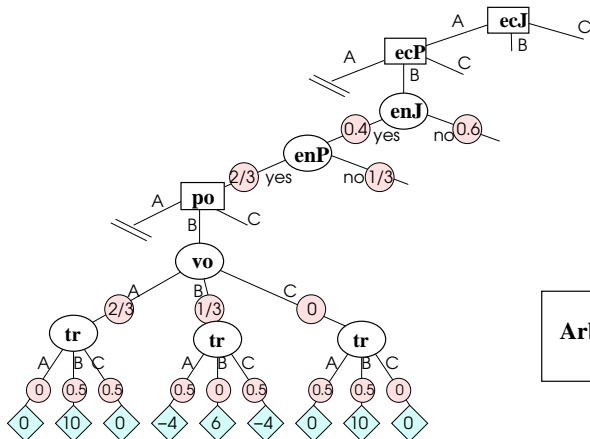
Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Conclusion

Requête sur le problème

Règles de décision **maximisant l'utilité espérée** si d'abord Pierre et Jean écoutent chacun à une porte, et ensuite Jean choisit une porte à ouvrir en fonction de ce qui a été entendu?



Arbre de décision

Requête sur le problème

Répondre à la requête en utilisant un arbre de décision



Calculer des règles de décision optimales pour la quantité

$$\max_{ec_J, ec_P} \sum_{en_J, en_P} \max_{po} \sum_{vo, tr} \left(\left(\bigwedge_{i \in [1,2]} F_i \right) \star \left(\prod_{i \in [1,4]} P_i \right) \times \left(\sum_{i \in [1,2]} U_i \right) \right)$$

Eliminer une variable x → ex: $\sum_x f_{x,y} = \sum_{a \in dom(x)} f_{x,y}(a) = g_y$
 $\max_x h_{x,y,z}$

...

\star = opérateur de troncature qui masque les décisions infaisables

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Conclusion

2^{ème} notion importante du cadre:

Prise de décision par calcul
de séquences d'éliminations de variables
sur des combinaisons de fonctions locales

Autres modèles de plausibilité/utilité

$$\max_{ec_p, ec_j} \sum_{en_j, en_p} \max_{po} \sum_{vo, tr} \left(\left(\bigwedge_{i \in [1,2]} F_i \right) \star \left(\prod_{i \in [1,4]} P_i \right) \times \left(\sum_{i \in [1,2]} U_i \right) \right)$$

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Conclusion

Autres modèles de plausibilité/utilité

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Conclusion

$$\max_{ec_P, ec_J} \sum_{en_J, en_P} \max_{po} \sum_{vo, tr} \left(\left(\bigwedge_{i \in [1,2]} F_i \right) \star \left(\prod_{i \in [1,4]} P_i \right) \times \left(\sum_{i \in [1,2]} U_i \right) \right)$$

$$\max_{ec_P, ec_J} \max_{en_J, en_P} \max_{po} \max_{vo, tr} \left(\left(\bigwedge_{i \in [1,2]} F_i \right) \star \min \left(\left(\min_{i \in [1,4]} P_i \right), \left(\min_{i \in [1,2]} U_i \right) \right) \right)$$

$$\max_{ec_P, ec_J} \min_{en_J, en_P} \max_{po} \min_{vo, tr} \left(\left(\bigwedge_{i \in [1,2]} F_i \right) \star \left(\bigwedge_{i \in [1,4]} P_i \right) \rightarrow \left(\bigvee_{i \in [1,2]} U_i \right) \right)$$

$$\max_{ec_P, ec_J} \min_{en_J, en_P} \max_{po} \min_{vo, tr} \left(\left(\bigwedge_{i \in [1,2]} F_i \right) \star \left(\bigwedge_{i \in [1,4]} P_i \right) \wedge \left(\bigwedge_{i \in [1,2]} U_i \right) \right)$$

$$\max_{ec_P, ec_J} \sum_{en_J, en_P} \max_{po} \sum_{vo, tr} \left(\left(\bigwedge_{i \in [1,2]} F_i \right) \star \left(\prod_{i \in [1,4]} P_i \right) \times \left(\prod_{i \in [1,2]} U_i \right) \right)$$

$$\max_{ec_P, ec_J} \sum_{en_J, en_P} \max_{po} \sum_{vo, tr} \left(\left(\bigwedge_{i \in [1,2]} F_i \right) \star \left(\prod_{i \in [1,4]} P_i \right) \times \left(\sum_{i \in [1,2]} U_i \right) \right)$$

$$\max_{ec_P, ec_J} \max_{en_J, en_P} \max_{po} \max_{vo, tr} \left(\left(\bigwedge_{i \in [1,2]} F_i \right) \star \min \left(\left(\min_{i \in [1,4]} P_i \right), \left(\min_{i \in [1,2]} U_i \right) \right) \right)$$

$$\max_{ec_P, ec_J} \min_{en_J, en_P} \max_{po} \min_{vo, tr} \left(\left(\bigwedge_{i \in [1,2]} F_i \right) \star \left(\bigwedge_{i \in [1,4]} P_i \right) \rightarrow \left(\bigvee_{i \in [1,2]} U_i \right) \right)$$

$$\max_{ec_P, ec_J} \min_{en_J, en_P} \max_{po} \min_{vo, tr} \left(\left(\bigwedge_{i \in [1,2]} F_i \right) \star \left(\bigwedge_{i \in [1,4]} P_i \right) \wedge \left(\bigwedge_{i \in [1,2]} U_i \right) \right)$$

$$\max_{ec_P, ec_J} \sum_{en_J, en_P} \max_{po} \sum_{vo, tr} \left(\left(\bigwedge_{i \in [1,2]} F_i \right) \star \left(\prod_{i \in [1,4]} P_i \right) \times \left(\prod_{i \in [1,2]} U_i \right) \right)$$

Forme générique:

$$\max_{ec_J, ec_P} \oplus_U \max_{en_J, en_P} \oplus_U \max_{po} \sum_{vo, tr} \left(\left(\bigwedge_{F_i \in F} F_i \right) \star \left(\bigotimes_{P_i \in P} P_i \right) \otimes_{pu} \left(\bigotimes_{U_i \in U} U_i \right) \right)$$

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Conclusion

3^{ème} notion importante du cadre:

Aspect algébrique utile
pour développer un cadre générique

Introduction

Exemple

Le cadre PFU

Structure
algébrique
générique

Réseau de
fonctions locales

Requêtes

Analyse du cadre

Algorithmes
génériques sur
le cadre PFU

Conclusion

1 Exemple introductif

2 Le cadre PFU

- Structure algébrique générique
- Réseau de fonctions locales
- Requêtes
- Analyse du cadre

3 Algorithmes génériques sur le cadre PFU

Introduction

Exemple

Le cadre PFU

Structure
algébrique
générique

Réseau de
fonctions locales

Requêtes

Analyse du cadre

Algorithmes
génériques sur
le cadre PFU

Conclusion

1 Exemple introductif

2 Le cadre PFU

- **Structure algébrique générique**
- Réseau de fonctions locales
- Requêtes
- Analyse du cadre

3 Algorithmes génériques sur le cadre PFU

Premier élément clé: une structure algébrique

Introduction

Exemple

Le cadre PFU

Structure
algébrique
générique

Réseau de
fonctions locales

Requêtes

Analyse du cadre

Algorithmes
génériques sur
le cadre PFU

Conclusion

But: spécifier comment les informations sont **combinées et synthétisées**

Plusieurs structures:

	Forme	Exemple
structure de plausibilité	$(E_p, \oplus_p, \otimes_p)$	$(\mathbb{R}^+, +, \times)$
structure d'utilité	(E_u, \otimes_u)	$(\mathbb{R}, +)$
structure d'utilité espérée	$(E_p, E_u, \oplus_u, \otimes_{pu})$	$(\mathbb{R}^+, \mathbb{R}, +, \times)$

Axiomes sur ces structures inspirés par [Friedman-Halpern' 95] et [Chu-Halpern' 03]

Introduction

Exemple

Le cadre PFU

Structure
algébrique
générique

**Réseau de
fonctions locales**

Requêtes

Analyse du cadre

Algorithmes
génériques sur
le cadre PFU

Conclusion

1 Exemple introductif

2 Le cadre PFU

- Structure algébrique générique
- **Réseau de fonctions locales**
- Requêtes
- Analyse du cadre

3 Algorithmes génériques sur le cadre PFU

Deuxième élément clé: réseau PFU

Introduction

Exemple

Le cadre PFU

Structure
algébrique
générique

Réseau de
fonctions locales

Requêtes

Analyse du cadre

Algorithmes
génériques sur
le cadre PFU

Conclusion

But: définir des **variables** et des **fonctions locales** portant sur ces variables

Définition

Un **réseau PFU** est un tuple (V, P, F, U, G) où

- V : ensemble fini de **variables de décision et d'environnement**
- $P = \{P_1, P_2, \dots\}$: ensemble fini de fonctions de **plausibilité**
- $F = \{F_1, F_2, \dots\}$: ensemble fini de fonctions de **faisabilité**
- $U = \{U_1, U_2, \dots\}$: ensemble fini de fonctions d'**utilité**
- G : DAG représentant des **conditions de normalisation**

Façon d'obtenir les réseaux justifiée via des résultats généralisant les liens entre **factorisation** et **indépendance conditionnelle**

Réseau PFU du problème du trésor

Introduction

Exemple

Le cadre PFU

Structure
algébrique
générique

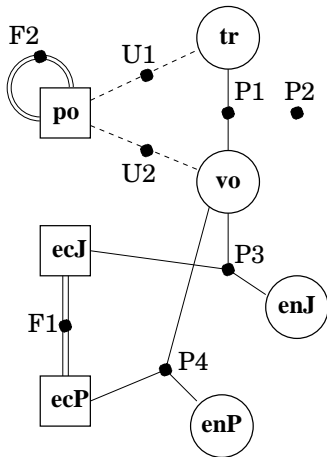
Réseau de
fonctions locales

Requêtes

Analyse du cadre

Algorithmes
génériques sur
le cadre PFU

Conclusion



Introduction

Exemple

Le cadre PFU

Structure
algébrique
générique

Réseau de
fonctions locales

Requêtes

Analyse du cadre

Algorithmes
génériques sur
le cadre PFU

Conclusion

1 Exemple introductif

2 Le cadre PFU

- Structure algébrique générique
- Réseau de fonctions locales
- **Requêtes**
- Analyse du cadre

3 Algorithmes génériques sur le cadre PFU

Troisième élément clé: requêtes sur un réseau PFU

Introduction

Exemple

Le cadre PFU

Structure

algébrique

générique

Réseau de
fonctions locales

Requêtes

Analyse du cadre

Algorithmes

génériques sur

le cadre PFU

Conclusion

But: définir des **problèmes de décision**

Définition

Requête = **séquence d'éliminations SE**

$$\text{ex: } SE = \max_{ec_J, ec_P} \oplus_{u_{en_J, en_P}} \max_{po} \oplus_{u_{vo, tr}}$$

$$SE = \min_{ec_P} \max_{ec_J} \oplus_{u_{en_J}} \max_{po} \oplus_{u_{en_P}} \oplus_{u_{vo, tr}}$$

Troisième élément clé: requêtes sur un réseau PFU

Introduction

Exemple

Le cadre PFU

Structure
algébrique
générique

Réseau de
fonctions locales

Requêtes

Analyse du cadre

Algorithmes
génériques sur
le cadre PFU

Conclusion

But: définir des **problèmes de décision**

Définition

Requête = **séquence d'éliminations SE**

$$\text{ex: } SE = \max_{ec_J, ec_P} \oplus_{u_{en_J}, en_P} \max_{po} \oplus_{u_{vo}, tr}$$

$$SE = \min_{ec_P} \max_{ec_J} \oplus_{u_{en_J}} \max_{po} \oplus_{u_{en_P}} \oplus_{u_{vo}, tr}$$

Sémantique:

- ordre d'élimination → **ordre des décisions et des observations**

Introduction

Exemple

Le cadre PFU

Structure

algébrique

générique

Réseau de
fonctions locales

Requêtes

Analyse du cadre

Algorithmes
génériques sur
le cadre PFU

Conclusion

But: définir des **problèmes de décision**

Définition

Requête = **séquence d'éliminations SE**

$$\text{ex: } SE = \max_{ec_J, ec_P} \oplus_{u_{en_J}, en_P} \max_{po} \oplus_{u_{vo, tr}}$$

$$SE = \min_{ec_P} \max_{ec_J} \oplus_{u_{en_J}} \max_{po} \oplus_{u_{en_P}} \oplus_{u_{vo, tr}}$$

Sémantique:

- ordre d'élimination → **ordre des décisions et des observations**
- min ou max sur une décision → attitude **optimiste ou pessimiste**

Troisième élément clé: requêtes sur un réseau PFU

Répondre à une requête = calculer des **utilités espérées optimales** et/ou des **règles de décision optimales**

Réponse à une requête Q :

$$Ans(Q) = SE \left(\left(\bigwedge_{F_i \in F} F_i \right) \star \left(\bigotimes_{P_i \in P} P_i \right) \otimes_{pu} \left(\bigotimes_{U_i \in U} U_i \right) \right)$$

Introduction

Exemple

Le cadre PFU

Structure

algébrique

générique

Réseau de
fonctions locales

Requêtes

Analyse du cadre

Algorithmes
génériques sur
le cadre PFU

Conclusion

Troisième élément clé: requêtes sur un réseau PFU

Répondre à une requête = calculer des **utilités espérées optimales** et/ou des **règles de décision optimales**

Réponse à une requête Q :

$$Ans(Q) = SE \left(\left(\bigwedge_{F_i \in F} F_i \right) \star \left(\bigotimes_{P_i \in P} P_i \right) \otimes_{pu} \left(\bigotimes_{U_i \in U} U_i \right) \right)$$

Théorème

La définition de $Ans(Q)$ (vision opérationnelle) est **équivalente** à une définition à base d'arbre de décision (vision sémantique)

Introduction

Exemple

Le cadre PFU

Structure

algébrique

générique

Réseau de
fonctions locales

Requêtes

Analyse du cadre

Algorithmes
génériques sur
le cadre PFU

Conclusion

Troisième élément clé: requêtes sur un réseau PFU

Répondre à une requête = calculer des **utilités espérées optimales** et/ou des **règles de décision optimales**

Réponse à une requête Q :

$$Ans(Q) = SE \left(\left(\bigwedge_{F_i \in F} F_i \right) \star \left(\bigotimes_{P_i \in P} P_i \right) \otimes_{pu} \left(\bigotimes_{U_i \in U} U_i \right) \right)$$

Théorème

La définition de $Ans(Q)$ (vision opérationnelle) est **équivalente** à une définition à base d'arbre de décision (vision sémantique)

Théorème

Savoir si $Ans(Q) > \alpha$ est un problème PSPACE-complet

Introduction

Exemple

Le cadre PFU

Structure

algébrique

générique

Réseau de

fonctions locales

Requêtes

Analyse du cadre

Algorithmes

génériques sur

le cadre PFU

Conclusion

Introduction

Exemple

Le cadre PFU

Structure
algébrique
générique

Réseau de
fonctions locales

Requêtes

Analyse du cadre

Algorithmes
génériques sur
le cadre PFU

Conclusion

Formalisme obtenu: **le cadre PFU (Plausibilité-Faisabilité-Utilité)**



PRALET, VERFAILLIE, SCHIEX. *An algebraic graphical model for decision with uncertainties, feasibilities, and utilities* - *Journal of Artificial Intelligence Research* (JAIR), 2007



PRALET, VERFAILLIE, SCHIEX. *Un cadre graphique et algébrique pour les problèmes de décision incluant incertitudes, faisabilités et utilités* - *Revue d'Intelligence Artificielle* (RIA), 2006



PRALET, VERFAILLIE, SCHIEX. *Decision with uncertainties, feasibilities, and utilities : towards a unified algebraic framework* - *European Conference on Artificial Intelligence* (ECAI), 2006

Introduction

Exemple

Le cadre PFU

Structure
algébrique
générique

Réseau de
fonctions locales

Requêtes

Analyse du cadre

Algorithmes
génériques sur
le cadre PFU

Conclusion

1 Exemple introductif

2 Le cadre PFU

- Structure algébrique générique
- Réseau de fonctions locales
- Requêtes
- **Analyse du cadre**

3 Algorithmes génériques sur le cadre PFU

Théorème d'unification

Théorème: le cadre PFU permet d'exprimer les requêtes sur:

CSP
Valued CSP
Quantified CSP
Mixed and probabilistic CSP
Stochastic CSP

Finite-horizon Markov decision processes (MDP)
probabilistic/possibilistic/with kappa rankings
completely or partially (POMDP) observable
factored or not

SAT
Quantified boolean formulas
Stochastic SAT
Extended stochastic SAT

Bayesian networks
Hybrid networks
Markov Random Fields
Chain graphs

Probability computation
MPE (Most Probable Explanation)
MAP (Maximum A Posteriori hyp.)

Influence diagrams
Valuation networks

Finite-horizon
STRIPS planning,
Conformant planning,
Probabilistic planning

Introduction

Exemple

Le cadre PFU

Structure
algébrique
générique

Réseau de
fonctions locales

Requêtes

Analyse du cadre

Algorithmes
génériques sur
le cadre PFU

Conclusion

Théorème: le cadre PFU permet d'exprimer les requêtes sur:

CSP
Valued CSP
Quantified CSP
Mixed and probabilistic CSP
Stochastic CSP

Finite-horizon Markov decision processes (MDP)
probabilistic/possibilistic/with kappa rankings
completely or partially (POMDP) observable
factored or not

SAT
Quantified boolean formulas
Stochastic SAT
Extended stochastic SAT

Bayesian networks
Hybrid networks
Markov Random Fields
Chain graphs

Probability computation
MPE (Most Probable Explanation)
MAP (Maximum A Posteriori hyp.)

Influence diagrams
Valuation networks

Finite-horizon
STRIPS planning,
Conformant planning,
Probabilistic planning

Mais existence de **cadres non couverts** (ex: fonctions de croyance) car compromis entre expressivité et efficacité des algorithmes à venir

Introduction

Exemple

Le cadre PFU

**Algorithmes
génériques sur
le cadre PFU**

Voie 1: adaptation
et généralisation de
l'existant

Voie 2: introduction
de nouvelles
techniques

Solver

Conclusion

1 Exemple introductif

2 Le cadre PFU

3 Algorithmes génériques sur le cadre PFU

- Voie 1: adaptation et généralisation de l'existant
- Voie 2: introduction de nouvelles techniques
- Solver

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

**Voie 1: adaptation
et généralisation de
l'existant**

Voie 2: introduction
de nouvelles
techniques

Solver

Conclusion

1 Exemple introductif

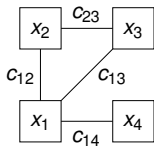
2 Le cadre PFU

3 Algorithmes génériques sur le cadre PFU

- **Voie 1: adaptation et généralisation de l'existant**
- Voie 2: introduction de nouvelles techniques
- Solver

Algorithme d'élimination de variables classique (VE)

Ex: calculer $q = \min_{x_1, x_2, x_3, x_4} (c_{12} + c_{13} + c_{23} + c_{14})$



Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Voie 1: adaptation
et généralisation de
l'existant

Voie 2: introduction
de nouvelles
techniques

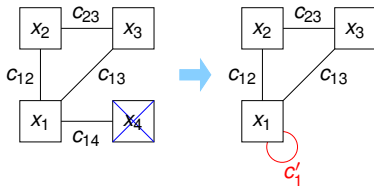
Solver

Conclusion

Algorithme d'élimination de variables classique (VE)

Ex: calculer $q = \min_{x_1, x_2, x_3, x_4} (c_{12} + c_{13} + c_{23} + c_{14})$

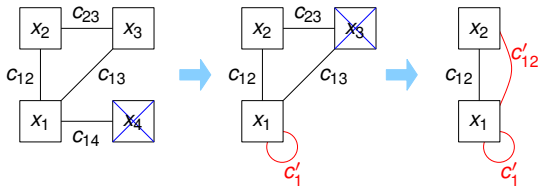
$$q = \min_{x_1, x_2, x_3} (c_{12} + c_{13} + c_{23} + \underbrace{(\min_{x_4} c_{14})}_{= c'_1})$$



Algorithme d'élimination de variables classique (VE)

Ex: calculer $q = \min_{x_1, x_2, x_3, x_4} (c_{12} + c_{13} + c_{23} + c_{14})$

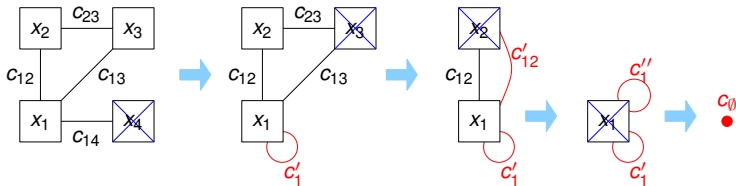
$$\begin{aligned} q &= \min_{x_1, x_2, x_3} (c_{12} + c_{13} + c_{23} + \underbrace{\min_{x_4} c_{14}}_{= c'_1}) \\ &= \min_{x_1, x_2} (c_{12} + (\min_{x_3} c_{13} + c_{23}) + c'_1) \end{aligned}$$



Algorithme d'élimination de variables classique (VE)

Ex: calculer $q = \min_{x_1, x_2, x_3, x_4} (c_{12} + c_{13} + c_{23} + c_{14})$

$$\begin{aligned} q &= \min_{x_1, x_2, x_3} (c_{12} + c_{13} + c_{23} + \underbrace{\min_{x_4} c_{14}}_{= c'_1}) \\ &= \min_{x_1, x_2} (c_{12} + (\min_{x_3} c_{13} + c_{23}) + c'_1) \dots \end{aligned}$$



Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Voie 1: adaptation
et généralisation de
l'existant

Voie 2: introduction
de nouvelles
techniques

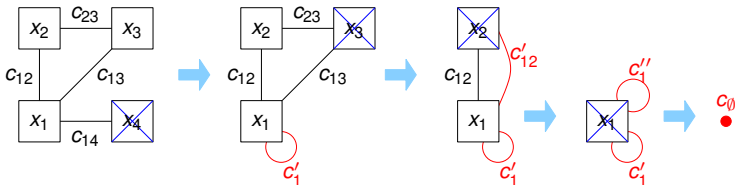
Solver

Conclusion

Algorithme d'élimination de variables classique (VE)

Ex: calculer $q = \min_{x_1, x_2, x_3, x_4} (c_{12} + c_{13} + c_{23} + c_{14})$

$$\begin{aligned} q &= \min_{x_1, x_2, x_3} (c_{12} + c_{13} + c_{23} + \underbrace{\min_{x_4} c_{14}}_{= c'_1}) \\ &= \min_{x_1, x_2} (c_{12} + (\min_{x_3} c_{13} + c_{23}) + c'_1) \end{aligned}$$



Décomposition d'un **calcul global** $\bigoplus_V \left(\bigotimes_{\varphi \in \Phi} \varphi \right)$ en **calculs locaux**

Complexité théorique exponentielle en la **largeur induite** (taille max des sous-problèmes résolus - 1)

Particularités du cadre PFU:

- 1 dans le cas PFU, pour éliminer une variable x , on ne peut pas toujours considérer seulement les fonctions dépendantes de x
- 2 plusieurs opérateurs de combinaison
- 3 plusieurs opérateurs d'élimination

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Voie 1: adaptation
et généralisation de
l'existant

Voie 2: introduction
de nouvelles
techniques

Solver

Conclusion

Particularités du cadre PFU:

- 1 dans le cas PFU, pour éliminer une variable x , on ne peut pas toujours considérer seulement les fonctions dépendantes de x
Identification de conditions suffisantes de "décomposabilité"
- 2 plusieurs opérateurs de combinaison
Transformations algébriques permettant de n'utiliser qu'un seul opérateur de combinaison
- 3 plusieurs opérateurs d'élimination
Contraintes sur l'ordre d'élim. (ex: $\max_{x,y} \sum_z \min_t \Rightarrow x, y \prec z \prec t$)

*Définition de l'algorithme **MVE** (Multi-operator-VE)*

Complexité théorique exponentielle en la largeur induite contrainte



PRALET, SCHIEX, VERFAILLIE. *Algorithmes et complexités génériques pour différents cadres de décision séquentielle dans l'incertain - Revue d'Intelligence Artificielle, 2007*

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Voie 1: adaptation
et généralisation de
l'existant

Voie 2: introduction
de nouvelles
techniques

Solver

Conclusion

1 Exemple introductif

2 Le cadre PFU

3 Algorithmes génériques sur le cadre PFU

- Voie 1: adaptation et généralisation de l'existant
- **Voie 2: introduction de nouvelles techniques**
- Solver

Certains aspects non utilisés par MVE

ex: **libertés cachées** dans l'ordre d'élimination

$$\sum_x \max_y \sum_z (\varphi_{xy} \cdot \varphi_{xz} \cdot \varphi_{xt}) = \sum_{xz} \max_y (\varphi_{xy} \cdot \varphi_{xz} \cdot \varphi_{xt})$$

Introduction de mécanismes d'analyse de la **structure des requêtes**



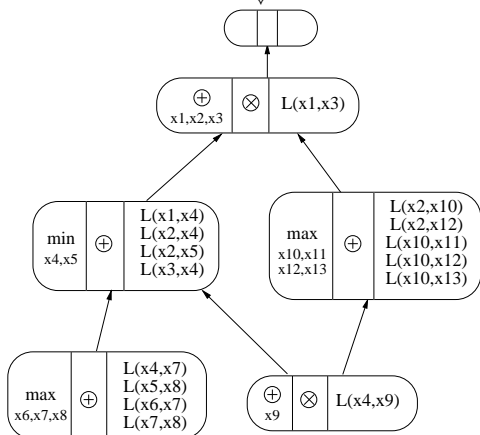
PRALET, SCHIEX, VERFAILLIE. *From influence diagrams to multi-operator cluster DAGs* - *International Conference on Uncertainty in Artificial Intelligence (UAI)*, 2006



PRALET, SCHIEX, VERFAILLIE. *Decomposition of multi-operator queries on semiring-based graphical models* - *International Conference on Principle and Practice of Constraint Programming (CP)*, 2006

Macro-structuration

$$SE \left(\left(\bigwedge_{F_i \in F} F_i \right) \star \left(\bigotimes_{P_i \in P} P_i \right) \otimes \left(\bigoplus_{U_i \in U} U_i \right) \right)$$



Requête
multi-opérateur
“plate”

Système de **règles
de réécriture**

DAG de requêtes
mono-opérateur

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Voie 1: adaptation
et généralisation de
l'existant

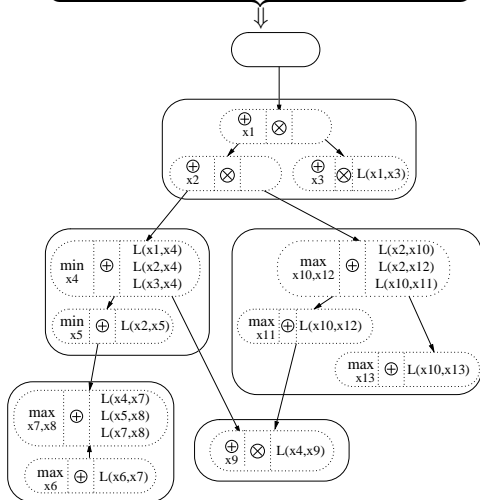
Voie 2: introduction
de nouvelles
techniques

Solver

Conclusion

Structuration plus fine

$$SE \left(\left(\bigwedge_{F_i \in F} F_i \right) * \left(\bigotimes_{P_i \in P} P_i \right) \otimes \left(\bigoplus_{U_i \in U} U_i \right) \right)$$



Requête
multi-opérateur
“plate”

Techniques de type
“**cluster-tree
decomposition**”

Architecture de calcul
MCDAG
(**M**ulti-opérateur
Cluster **D**AG)

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Voie 1: adaptation
et généralisation de
l'existant

Voie 2: introduction
de nouvelles
techniques

Solver

Conclusion

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Voie 1: adaptation
et généralisation de
l'existant

Voie 2: introduction
de nouvelles
techniques

Solver

Conclusion

Théorème

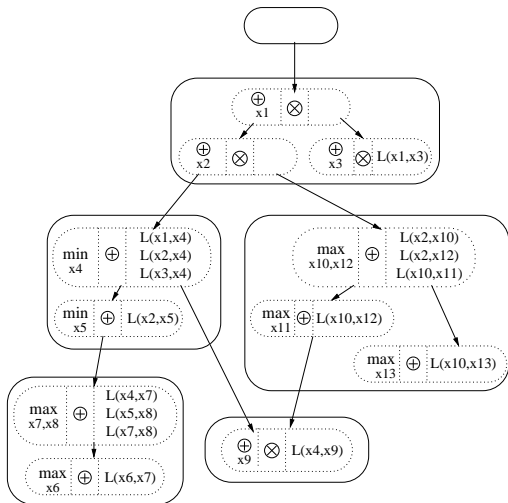
Le processus de réécriture (macrostructuration) est **correct et complet**.
Il conduit à une macrostructure **unique**. Il se fait en **temps polynomial**.

Théorème

La structuration **ne change pas ou fait diminuer** la largeur induite.
Elle **peut générer des gains exponentiels** de complexité théorique.

Réalisation de **tests** montrant des diminutions notoires de largeur induite
(d'un facteur 1 à 10) sur des exemples de formules booléennes
quantifiées de la librairie **QBF**

Utilisation de l'architecture MCDAG: approche "bottom-up"



Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

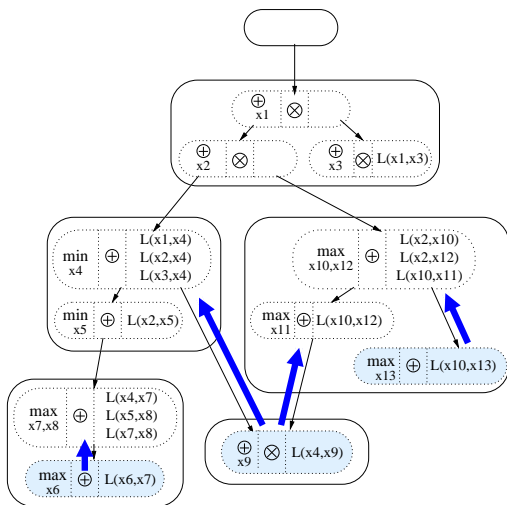
Voie 1: adaptation
et généralisation de
l'existant

Voie 2: introduction
de nouvelles
techniques

Solver

Conclusion

Utilisation de l'architecture MCDAG: approche "bottom-up"



Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

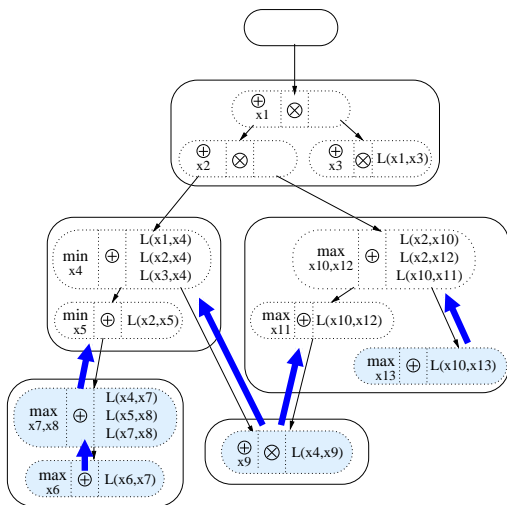
Voie 1: adaptation
et généralisation de
l'existant

Voie 2: introduction
de nouvelles
techniques

Solver

Conclusion

Utilisation de l'architecture MCDAG: approche "bottom-up"



Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

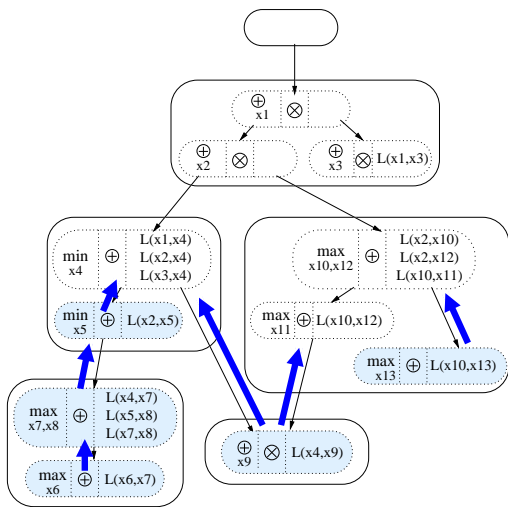
Voie 1: adaptation
et généralisation de
l'existant

Voie 2: introduction
de nouvelles
techniques

Solver

Conclusion

Utilisation de l'architecture MCDAG: approche "bottom-up"



Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

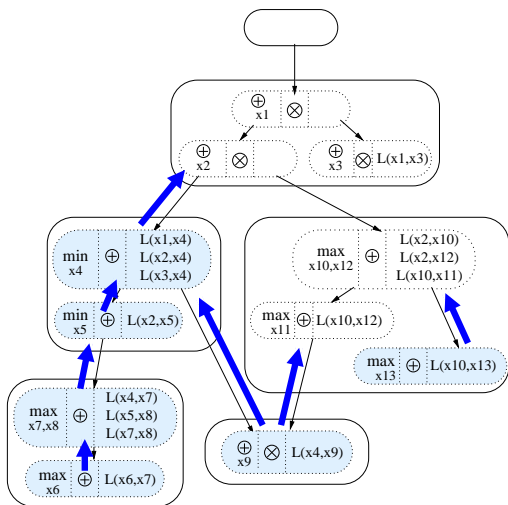
Voie 1: adaptation
et généralisation de
l'existant

Voie 2: introduction
de nouvelles
techniques

Solver

Conclusion

Utilisation de l'architecture MCDAG: approche "bottom-up"



Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

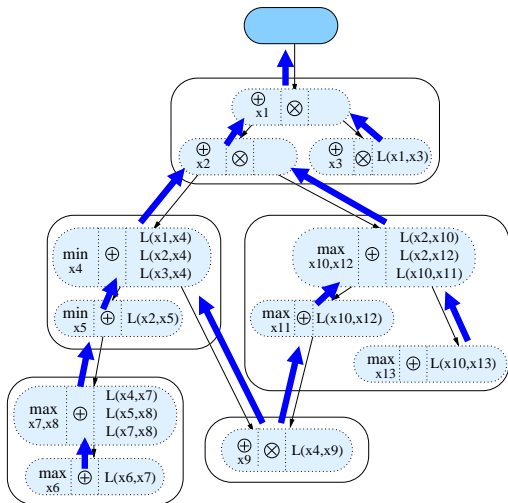
Voie 1: adaptation
et généralisation de
l'existant

Voie 2: introduction
de nouvelles
techniques

Solver

Conclusion

Utilisation de l'architecture MCDAG: approche "bottom-up"



Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

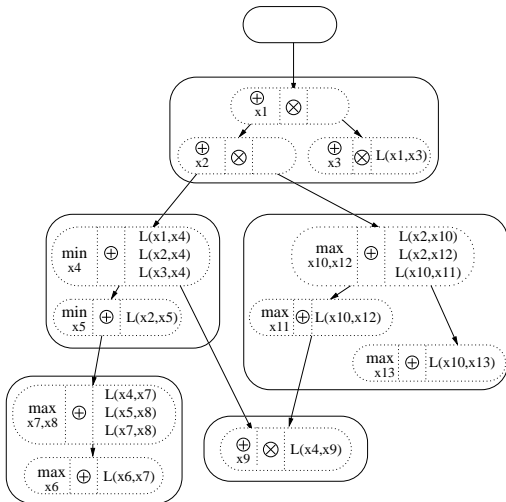
Voie 1: adaptation
et généralisation de
l'existant

Voie 2: introduction
de nouvelles
techniques

Solver

Conclusion

Utilisation de l'architecture MCDAG: approche "top-down"



Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Voie 1: adaptation
et généralisation de
l'existant

Voie 2: introduction
de nouvelles
techniques

Solver

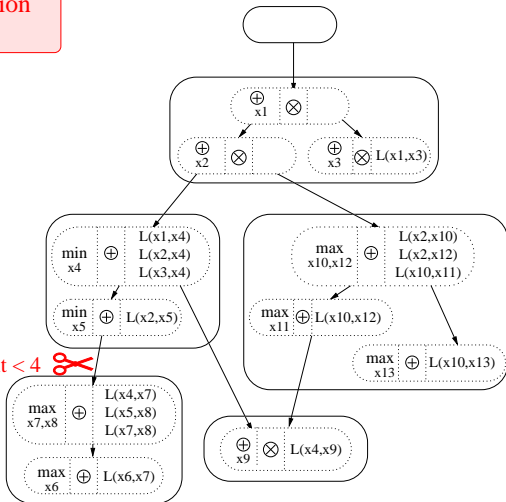
Conclusion

Utilisation de l'architecture MCDAG: approche "top-down"

✂ Bornes et propagation de contraintes



Exigence : $2 < \text{result} < 4$ ✂



Utilisation de l'architecture MCDAG: approche "top-down"

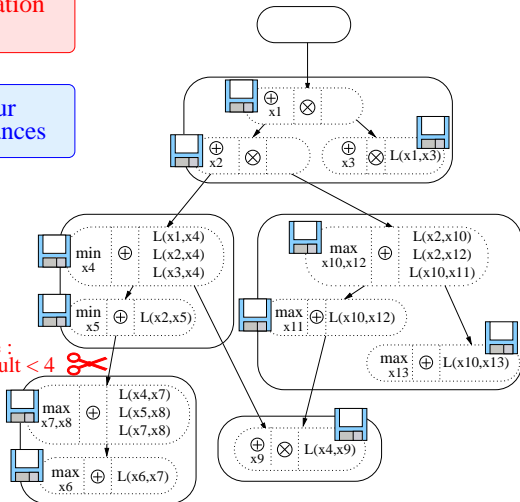


Bornes et propagation de contraintes



Mémorisation pour éviter les redondances

Exigence :
 $2 < \text{result} < 4$



Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Voie 1: adaptation
et généralisation de
l'existant

Voie 2: introduction
de nouvelles
techniques

Solver

Conclusion

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

Voie 1: adaptation
et généralisation de
l'existant

Voie 2: introduction
de nouvelles
techniques

Solver

Conclusion

1 Exemple introductif

2 Le cadre PFU

3 Algorithmes génériques sur le cadre PFU

- Voie 1: adaptation et généralisation de l'existant
- Voie 2: introduction de nouvelles techniques
- **Solver**

Introduction

Exemple

Le cadre PFU

Algorithmes
génériques sur
le cadre PFU

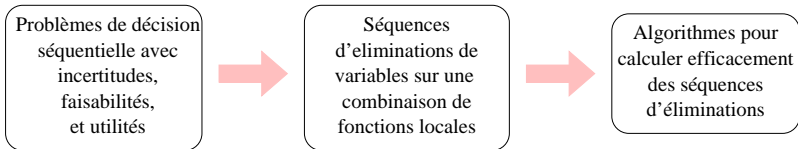
Voie 1: adaptation
et généralisation de
l'existant

Voie 2: introduction
de nouvelles
techniques

Solver

Conclusion

- Outil de résolution **générique** en C++
- Problèmes décrits en utilisant un format **XML**, ou des **formats existants** pour les QBF, les réseaux bayésiens ou les diagrammes d'influence
- Formes limitées de **propagation de contraintes**
- Quelques **tests** (pb de déploiement de constellation de satellite)



Utilisation d'une **approche algébrique** pour développer un **cadre unifié**

Algorithmes **génériques** exploitant des propriétés algébriques

Aspect composite des connaissances manipulées