

# Logique déontique pour la spécification du concept de disponibilité

Julien Brunel

Institut de Recherche en Informatique de Toulouse

Mai 2005 / Projet DISPO

# Plan

- 1 Introduction
  - Contexte
  - Logique déontique
  - Ajout du temps
- 2 Vers une logique d'action étendue
  - Logique dynamique
  - Logique déontique dynamique
- 3 Cas d'étude
  - Allocateur de ressources
  - Expression en logique dynamique étendue
- 4 Conclusion & Perspectives

# Plan

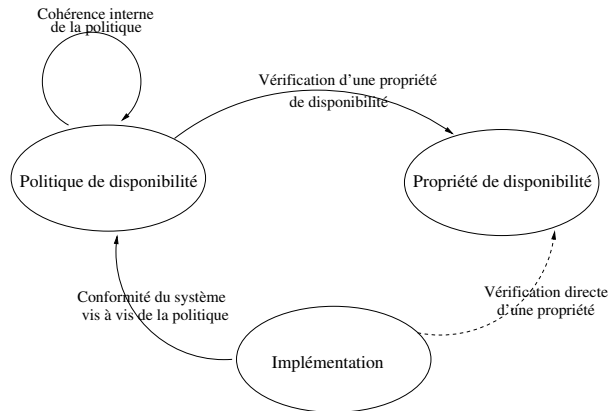
- 1 Introduction
  - Contexte
    - Logique déontique
    - Ajout du temps
- 2 Vers une logique d'action étendue
  - Logique dynamique
  - Logique déontique dynamique
- 3 Cas d'étude
  - Allocateur de ressources
  - Expression en logique dynamique étendue
- 4 Conclusion & Perspectives

# Disponibilité

## Concept clé en sécurité informatique

- Modèle de Saurel et Cuppens
- ⇒ Capacité à gérer les demandes de ressource avec des contraintes de temps
- Concept de politique de disponibilité
- Concept de propriété de disponibilité

# Problématique



# Plan

- 1 Introduction
  - Contexte
  - **Logique déontique**
  - Ajout du temps
- 2 Vers une logique d'action étendue
  - Logique dynamique
  - Logique déontique dynamique
- 3 Cas d'étude
  - Allocateur de ressources
  - Expression en logique dynamique étendue
- 4 Conclusion & Perspectives

# Logique déontique

But : exprimer

- Obligation  $O(\phi)$
- Interdiction  $I(\phi) \stackrel{def}{=} O(\neg\phi)$
- Permission  $P(\phi) \stackrel{def}{=} \neg I(\phi)$

# Logique déontique

But : exprimer

- Obligation  $O(\phi)$
- Interdiction  $I(\phi) \stackrel{def}{=} O(\neg\phi)$
- Permission  $P(\phi) \stackrel{def}{=} \neg I(\phi)$

# Logique déontique

But : exprimer

- Obligation  $O(\phi)$
- Interdiction  $I(\phi) \stackrel{def}{=} O(\neg\phi)$
- Permission  $P(\phi) \stackrel{def}{=} \neg I(\phi)$

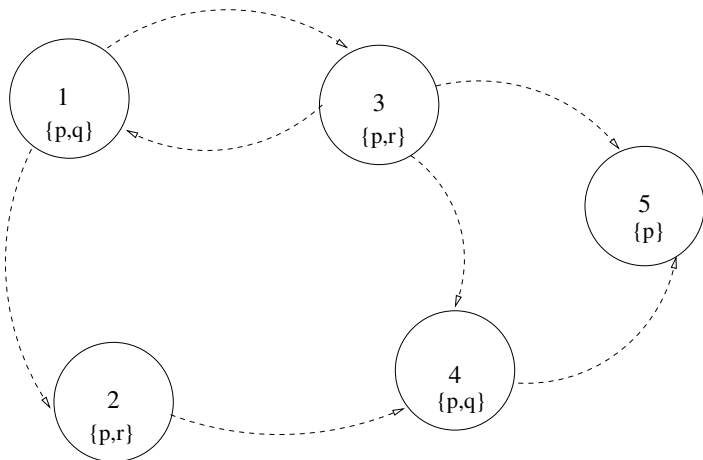
# Axiomes de *SDL*

- $O(\phi \Rightarrow \psi) \Rightarrow (O(\phi) \Rightarrow O(\psi))$
- $O(\phi) \Rightarrow P(\phi)$

$$\frac{\phi, \phi \Rightarrow \psi}{\psi} \text{ Modus Ponens}$$

$$\frac{\phi}{O(\phi)} \text{ O-nécessitation}$$

# Sémantique de *SDL*



# Raisonner dans *SDL*

- $conduire \stackrel{def}{=} \text{“Je conduis”}$
- $demarrer \stackrel{def}{=} \text{“J’ai démarré”}$

$$P(conduire) \wedge O(conduire \Rightarrow demarrer) \Rightarrow P(demarrer)$$

$$I(demarrer) \wedge O(conduire \Rightarrow demarrer) \Rightarrow I(conduire)$$

## Rappel

- $I$  Interdiction
- $P$  Permission

# Plan

- 1 Introduction
  - Contexte
  - Logique déontique
  - Ajout du temps
- 2 Vers une logique d'action étendue
  - Logique dynamique
  - Logique déontique dynamique
- 3 Cas d'étude
  - Allocateur de ressources
  - Expression en logique dynamique étendue
- 4 Conclusion & Perspectives

# Besoin du temps

- Le processus  $p_1$  a toujours la permission d'utiliser la ressource
  - Le processus  $p_1$  est obligé de libérer la ressource avant 3 unités de temps à partir du moment où il y a accédé
- Logique temporelle / temporisée
- On se base sur *CTL*
- **On ajoute une relation déontique** (qui donne une sémantique à l'obligation)

# Besoin du temps

- Le processus  $p_1$  a toujours la permission d'utiliser la ressource
- Le processus  $p_1$  est obligé de libérer la ressource avant 3 unités de temps à partir du moment où il y a accédé

→ Logique temporelle / temporisée

→ On se base sur *CTL*

→ **On ajoute une relation déontique** (qui donne une sémantique à l'obligation)

# Besoin du temps

- Le processus  $p_1$  a toujours la permission d'utiliser la ressource
  - Le processus  $p_1$  est obligé de libérer la ressource avant 3 unités de temps à partir du moment où il y a accédé
- Logique temporelle / temporisée
- On se base sur *CTL*
- **On ajoute une relation déontique** (qui donne une sémantique à l'obligation)

# Besoin du temps

- Le processus  $p_1$  a toujours la permission d'utiliser la ressource
  - Le processus  $p_1$  est obligé de libérer la ressource avant 3 unités de temps à partir du moment où il y a accédé
- 
- Logique temporelle / temporisée
  - On se base sur *CTL*
  - **On ajoute une relation déontique** (qui donne une sémantique à l'obligation)

# CTL + déontique

## Mélange déontique/temporel

Quelles propriétés relient obligation et temps ?

- $O(A\Box \phi) \stackrel{?}{\Rightarrow} A\Box O(\phi)$
- $P(A\Box \phi) \stackrel{?}{\Rightarrow} A\Box P(\phi)$
- $O(\phi)$  avec  $\phi$  **non** temporelle

Un tel choix implique des propriétés sur la relation déontique

## Lien entre les relations temporelle et déontique

$\Rightarrow$  Analogie avec une relation de raffinement

# Politiques de disponibilité

- Si  $s_1$  demande la ressource et  $s_2$  ne l'utilise pas, alors  $s_1$  est autorisé à l'utiliser

$$A\Box (demande(s_1, r) \wedge \neg utilise(s_2, r) \Rightarrow P utilise(s_1, r))$$

- Si  $s_i$  demande la ressource et a la permission de l'utiliser, alors il a l'obligation de l'utiliser avant 10 unités de temps.

$$A\Box (demande(s_i, r) \wedge P utilise(s_1, r) \Rightarrow O_{\leq 10} utilise(s_1, r))$$

- $s_i$  a l'interdiction d'utiliser la ressource pendant 6 unités de temps après l'avoir libérée

$$A\Box (libere(s_i, r) \Rightarrow A\Box_{\leq 6} I(utilise(s_i, r)))$$

# Que veut-on vérifier ?

- Vérifier que la politique est consistente
  - ⇒ On doit être capable de décider la satisfiabilité de la formule correspondante (méthode des tableaux)
- Vérifier que le système respecte la politique
  - ⇒ On enrichit le modèle du système par une relation déontique
  - ⇒ On vérifie des propriétés sur le modèle obtenu

# Plan

- 1 Introduction
  - Contexte
  - Logique déontique
  - Ajout du temps
- 2 Vers une logique d'action étendue
  - **Logique dynamique**
  - Logique déontique dynamique
- 3 Cas d'étude
  - Allocateur de ressources
  - Expression en logique dynamique étendue
- 4 Conclusion & Perspectives

# Logique dynamique

Besoin d'exprimer l'obligatoir de faire, et l'obligation d'être  
 $\Rightarrow$  On s'inspire des logiques d'action.

- ensemble d'actions atomiques  $\Sigma = \{a, b \dots\}$
- ensemble d'actions  $\mathcal{A} \ni \alpha, \beta \stackrel{def}{=} a \mid \alpha + \beta \mid \alpha; \beta \mid \alpha^*$

## Opérateurs de la logique dynamique

$\langle \alpha \rangle \phi$  Il existe un état accessible en exécutant  $\alpha$  qui satisfait  $\phi$ .

$[\alpha] \phi$  Tous les états accessibles en exécutant  $\alpha$  satisfont  $\phi$

# Logique temporelle étendue par les actions

## Until étendu par des actions

$E(\phi \mathcal{U}^{\alpha} \psi)$  signifie qu'il existe une exécution de  $\alpha$  à partir de l'état courant qui mène à un état qui satisfait  $\psi$ , et les états traversés satisfont  $\phi$ .

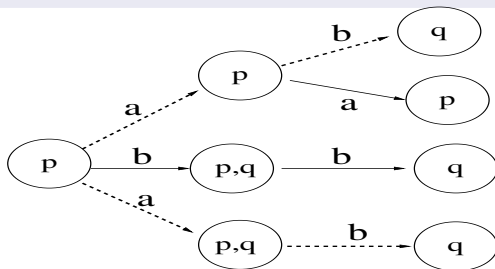


FIG.:  $A p \mathcal{U}^{a;b} q$

# Plan

- 1 Introduction
  - Contexte
  - Logique déontique
  - Ajout du temps
- 2 Vers une logique d'action étendue
  - Logique dynamique
  - **Logique déontique dynamique**
- 3 Cas d'étude
  - Allocateur de ressources
  - Expression en logique dynamique étendue
- 4 Conclusion & Perspectives

# Modèle déontique dynamique

## Langage

*Prop* propositions atomiques

$\Sigma$  et  $\mathcal{A}$  actions resp. atomiques et complexes

$E \phi \mathcal{U}^\alpha \psi$ ,  $A \phi \mathcal{U}^\alpha \psi$ ,  $O(\phi)$ .

## Modèle

- $W$  ensemble d'états
- $\Pi : W \rightarrow 2^{Prop}$  valuation
- $R_o \subseteq W \times W$  relation déontiques
- $\delta \subseteq W \times \Sigma \times W$  relation de transition

# Opérateurs supplémentaires

## Quelques opérateurs dynamiques temporels

$$\langle \alpha \rangle \phi \stackrel{\text{def}}{=} E \top U^\alpha \phi$$

$$[\alpha] \phi \stackrel{\text{def}}{=} \neg \langle \alpha \rangle \neg \phi$$

$$\diamond \alpha \stackrel{\text{def}}{=} \Sigma^*; \alpha$$

$$E \diamond \alpha \stackrel{\text{def}}{=} E \top U^{\diamond \alpha} \top$$

## Opérateurs déontiques

$$\mathcal{O}(\alpha) \stackrel{\text{def}}{=} \mathcal{O}(A \top U^\alpha \top)$$

$$\mathcal{I}(\alpha) \stackrel{\text{def}}{=} \mathcal{I}(E \top U^\alpha \top)$$

$$\mathcal{P}(\alpha) \stackrel{\text{def}}{=} \neg \mathcal{I}(\alpha)$$

# Plan

- 1 Introduction
  - Contexte
  - Logique déontique
  - Ajout du temps
- 2 Vers une logique d'action étendue
  - Logique dynamique
  - Logique déontique dynamique
- 3 **Cas d'étude**
  - **Allocateur de ressources**
  - Expression en logique dynamique étendue
- 4 Conclusion & Perspectives

# Allocateur de ressources

## Principe

- Des utilisateurs demandent à consommer une/des ressource(s)
- Définir une politique qui restreint les comportements

## Modélisation de Nora, Frédéric, Ahmed

- Prédicats  $req(cons)$ ,  $start(cons)$ ,  $done(cons)$
- Prédicats déontiques  
 $Droit_x(cons)$ ,  $O-start_x(cons)$ ,  $O-done_x(cons)$  où  $x \in \{b, f\}$

# Politique et propriétés de Nora

## Politique de disponibilité

- $P(\text{cons})$
- $\square \text{Droit}_f(\text{cons})$
- $\square \neg \text{O-start}_f(\text{cons})$
- $\square (\text{start}(\text{cons}) \Leftrightarrow \text{O-done}(\text{cons}))$

## Propriétés de disponibilité

- $\square \text{Droit}_f(\text{cons}) \wedge \text{req}(\text{cons}) \Rightarrow \diamond \text{start}(\text{cons})$  (P4)
- $\square \text{O-start}_f(\text{cons}) \Rightarrow \diamond \text{start}(\text{cons})$  (P6)
- $\square \text{O-done}_f(\text{cons}) \Rightarrow \diamond \text{done}(\text{cons})$  (P8)

# Plan

- 1 Introduction
  - Contexte
  - Logique déontique
  - Ajout du temps
- 2 Vers une logique d'action étendue
  - Logique dynamique
  - Logique déontique dynamique
- 3 **Cas d'étude**
  - Allocateur de ressources
  - **Expression en logique dynamique étendue**
- 4 Conclusion & Perspectives

# Allocation revisitée

## Traduction en logique dynamique étendue

But : utiliser un formalisme déontique et temporel pour être

- plus libre dans l'expressivité des politiques
- plus générique dans l'expression des propriétés de disponibilités

# L'allocation revisitée

Exprimons les prédicats  $Droit_f$ ,  $O\text{-done}_f$ ,  $O\text{-start}_f$  en logique dynamique étendue.

## Définition du langage

Actions atomiques  $\Sigma$  :  $req_{u,r}$ ,  $start_{u,r}$ ,  $done_{u,r}$

Propositions atomiques  $Prop$  :  $utilise_{u,r}$

## Traduction des prédicats en terme de modalités déontiques

$$Droit_f(cons) \stackrel{def}{=} P(utilise_{u,r}) \wedge req_{u,r} \Rightarrow \mathcal{O}(\diamond start_{u,r})$$
$$O\text{-start}_f(cons) \stackrel{def}{=} \mathcal{O}(\diamond start_{u,r})$$
$$O\text{-done}_f(cons) \stackrel{def}{=} \mathcal{O}(\diamond done_{u,r})$$

# L'allocation revisitée

La politique de disponibilité devient

## Politique

- $A \Box P(\text{utilise}_{u,r})$
- $A \Box (\text{req} \Rightarrow \mathcal{O}(\diamond \text{start}_{u,r}))$
- $A \Box (\neg \mathcal{O}(\diamond \text{start}_{u,r}))$
- $A \Box (\text{start}_{u,r} \Leftrightarrow \mathcal{O}(\diamond \text{done}_{u,r}))$

Les propriétés de disponibilités sont devenues de la forme

$$\mathcal{O}(\diamond \alpha) \Rightarrow \diamond \alpha$$

# Vérification du respect de la politique

On aimerait ne vérifier que

$A\Box(Op \Rightarrow p)$  et  $A\Box(Oa \Rightarrow a)$  pour  $a \in \Sigma$  et  $p \in Prop$   
(atomiques)

et pouvoir en déduire

$A\Box(O\phi \Rightarrow \phi)$  et  $A\Box(O\alpha \Rightarrow \alpha)$  pour  $\phi \in DDTL$  et  $\alpha \in \mathcal{A}$   
(quelconques).

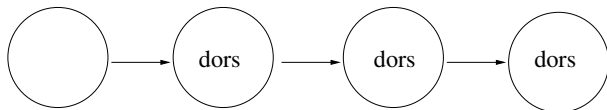
$\Rightarrow$  Impossible

# Vérification du respect de la politique

## Illustration

Si j'ai aujourd'hui l'obligation de dormir avant trois jours, peut-être n'aurai-je jamais l'obligation de dormir dans l'instant.

$$\nexists O(A \diamond_{\leq 3 \text{ jours}} \text{dormir}) \Rightarrow A \diamond_{\leq 3 \text{ jours}} O(\text{dormir})$$



$$O(A \diamond_{\leq 3} \text{dors}) \quad O(A \diamond_{\leq 2} \text{dors})$$

# Vérification

## Cohérence de la politique

### Méthode des tableaux (à développer)

## Respect de la politique de la part d'un système

- Trouver une modélisation d'un système
- Enrichir la modélisation pour tenir compte des obligations ?
- Ne tenir compte que des propositions et actions atomiques ? ( $\mathcal{O}p \Rightarrow p$  et  $\mathcal{O}a \Rightarrow a$ )
- Traduire dans une logique non déontique ?

# Conclusion & Perspectives

## Conclusion

- Étude de la logique déontique
- Lien avec les logiques temporelle et dynamique
- Spécification d'une politique de disponibilité

## Perspectives

- Composition , systèmes ouverts
- Consistence d'une politique
- Procédures de décision
- Environnement de preuve pour notre formalisme