Cours M2 'Ontologies et web sémantique' :

Introduction aux logiques de description

Andreas Herzig
Université de Toulouse, IRIT, CNRS
http://www.irit.fr/~Andreas.Herzig

(materiel en grande partie repris du cours de Nicola Olivetti,
Université Paul Cezanne, http://www.lsis.org/olivetti)

Master informatique
Université de Toulouse
2011-2012

Outline

DL = *description logics* = logiques de description

- deux buts :
  - représenter les connaissances
  - raisonner à partir de ces connaissances

Introduction : références

- *la* référence standard :
  - F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi and P. F. Patel-Schneider (eds.) “The Description Logic Handbook: Theory, Implementation, and Applications”.
    Cambridge University Press, 2003. Seconde édition en 2007.
- *la* resource web :
  - `http://dl.kr.org`

    $\Rightarrow$ DL workshop

    $\Rightarrow$ conférences

    $\Rightarrow$ ‘navigateur de complexité’

Introduction : quelques sigles

- DL = *description logics* = logiques de description
- KB = *knowledge base* = base de connaissances

$$\boxed{\text{KB} = \text{TBox} \cup \text{ABox}}$$

- TBox = *terminological box*
  = ontologie
  = connaissances non contingentes
- ABox = *assertion box*
  = connaissances contingentes

Introduction : quelques sigles

- DL = *description logics* = logiques de description
- KB = *knowledge base* = base de connaissances

$$\boxed{\text{KB} = \text{TBox} \cup \text{ABox}}$$

- TBox = *terminological box*
  = ontologie
  = connaissances non contingentes
- ABox = *assertion box*
  = connaissances contingentes

Introduction : la ABox

ABox = *assertion box*

- spécification des propriétés des individus (= prédicats unaires)
  - `Femelle(Anne)`
  - `Mâle(Charles)`
- spécification des relations binaires entre individus (= prédicats binaires)
  - `mèreDe(Anne,Charles)`
  - `pèreDe(Bob,Charles)`
  - `pèreDe(Charles,Dan)`
- la ABox utilise ('instancie') les propriétés et relations définies dans la TBox ...

Introduction : la ABox

ABox = *assertion box*

- spécification des propriétés des individus (= prédicats unaires)
  - `Femelle(Anne)`
  - `Mâle(Charles)`
- spécification des relations binaires entre individus (= prédicats binaires)
  - `mèreDe(Anne,Charles)`
  - `pèreDe(Bob,Charles)`
  - `pèreDe(Charles,Dan)`
- la ABox utilise ('instancie') les propriétés et relations définies dans la TBox . . .

Introduction : la TBox

TBox = *terminological box*

- définitions de propriétés et de relations entre propriétés
  - terminologie DL: concepts et rôles
- principalement : inclusion de concepts
  - $\mathtt{Mère} \sqsubseteq \mathtt{Parent}$
  - $\mathtt{Mère} \sqsubseteq \exists \mathtt{mèreDe}.\top$
  - $\exists \mathtt{mèreDe}.\top \sqsubseteq \mathtt{Mère}$
  - $\exists \mathtt{mèreDe}.(\exists \mathtt{mèreDe}.\top) \sqsubseteq \mathtt{GrandMère}$
- concepts primitifs
  - $\mathtt{Mère}$, $\mathtt{Père}$, $\mathtt{parentDe}$
- concepts complexes (non-primitifs)
  - $\mathtt{Oncle}$, $\mathtt{GrandOncle}$, $\mathtt{ArrièreCousin}$, $\mathtt{BelleSœur}$

  $\Rightarrow$ définis dans la TBox en termes d'autres concepts (primitifs ou complexes)

Introduction : les briques de base d'une TBox

- concepts
  - concepts atomiques : dénotent des ensembles d'individus
    - $\mathtt{Masculin}$, $\mathtt{Féminine}$, $\mathtt{Mère}$, ...
    - convention : commencent par majuscule
  - constructeurs de concepts ⇒ concepts complexes
    - opérateurs booléens : $\neg C$, $C \sqcap C$, $C \sqcup C$
    - opérateurs combinant concepts et rôles : $\exists R.C$, $\forall R.C$
- rôles
  - 'rôles' atomiques : dénotent des *relations binaires* entre individus
    - $\mathtt{mèreDe}$, $\mathtt{partieDe}$, $\mathtt{mange}$, ...
    - convention : commencent par minuscule
  - constructeurs de rôles ⇒ rôles complexes
    - opérateurs booléens : $\neg R$, $R \sqcap R$, $R \sqcup R$
    - opérateurs de l'algèbre des relations : $R \circ R$, R^*, R^{-1}, ...

choix des constructeurs ⇒ DL particulière

Introduction : les briques de base d'une TBox

- concepts
  - concepts atomiques : dénotent des ensembles d'individus
    - `Masculin`, `Féminine`, `Mère`, ...
    - convention : commencent par majuscule
  - constructeurs de concepts ⇒ concepts complexes
    - opérateurs booléens : $\neg C$, $C \sqcap C$, $C \sqcup C$
    - opérateurs combinant concepts et rôles : $\exists R.C$, $\forall R.C$
- rôles
  - 'rôles' atomiques : dénotent des *relations binaires* entre individus
    - `mèreDe`, `partieDe`, `mange`, ...
    - convention : commencent par minuscule
  - constructeurs de rôles ⇒ rôles complexes
    - opérateurs booléens : $\neg R$, $R \sqcap R$, $R \sqcup R$
    - opérateurs de l'algèbre des relations : $R \circ R$, R^*, R^{-1}, ...

choix des constructeurs ⇒ DL particulière

Introduction : les briques de base d'une TBox

- concepts
  - concepts atomiques : dénotent des ensembles d'individus
    - $\mathtt{Masculin}$, $\mathtt{Féminine}$, $\mathtt{Mère}$, ...
    - convention : commencent par majuscule
  - constructeurs de concepts $\Rightarrow$ concepts complexes
    - opérateurs booléens : $\neg C$, $C \sqcap C$, $C \sqcup C$
    - opérateurs combinant concepts et rôles : $\exists R.C$, $\forall R.C$

- rôles
  - 'rôles' atomiques : dénotent des *relations binaires* entre individus
    - $\mathtt{mèreDe}$, $\mathtt{partieDe}$, $\mathtt{mange}$, ...
    - convention : commencent par minuscule
  - constructeurs de rôles $\Rightarrow$ rôles complexes
    - opérateurs booléens : $\neg R$, $R \sqcap R$, $R \sqcup R$
    - opérateurs de l'algèbre des relations : $R \circ R$, R^*, R^{-1}, ...

$\boxed{\text{choix des constructeurs} \Rightarrow \text{DL particulière}}$

Introduction : un mot sur la terminologie

- rôles en DL (et donc dans la partie DL du cours) = relations binaires = prédicats binaires
  - rôles dans le sens ontologique = propriétés anti-rigides = prédicats unaires anti-rigides
    - `Mère`, `Père` sont des *rôles dans le sens ontologique* (définis à partir des relations binaires `mèreDe` et `pèreDe`)
    - à distinguer des concepts comme `Mâle` et `Femelle`

- concepts en DL = *propriétés* dans le sens ontologique
  - le sens ontologique du terme 'concept' est plus large

- dans les DL, pas de distinction entre
  - *catégories* = propriétés rigides (`Femelle`, `Mâle`)
  - *rôles* (dans le sens ontologique) = propriétés anti-rigides (`Mère`, `Enfant`, `Riche`)

  $\Rightarrow$ ontologie = 'photo' à un instant temporel donné

Introduction : un mot sur la terminologie

- rôles en DL (et donc dans la partie DL du cours) = relations binaires = prédicats binaires
  - rôles dans le sens ontologique = propriétés anti-rigides = prédicats unaires anti-rigides
    - `Mère`, `Père` sont des *rôles dans le sens ontologique* (définis à partir des relations binaires `mèreDe` et `pèreDe`)
    - à distinguer des concepts comme `Mâle` et `Femelle`
- concepts en DL = *propriétés* dans le sens ontologique
  - le sens ontologique du terme 'concept' est plus large
- dans les DL, pas de distinction entre
  - *catégories* = propriétés rigides (`Femelle`, `Mâle`)
  - *rôles* (dans le sens ontologique) = propriétés anti-rigides (`Mère`, `Enfant`, `Riche`)

  $\Rightarrow$ ontologie = 'photo' à un instant temporel donné

Introduction : un mot sur la terminologie

- rôles en DL (et donc dans la partie DL du cours) = relations binaires = prédicats binaires
  - rôles dans le sens ontologique = propriétés anti-rigides = prédicats unaires anti-rigides
    - `Mère`, `Père` sont des *rôles dans le sens ontologique* (définis à partir des relations binaires `mèreDe` et `pèreDe`)
    - à distinguer des concepts comme `Mâle` et `Femelle`

- concepts en DL = *propriétés* dans le sens ontologique
  - le sens ontologique du terme ‘concept’ est plus large

- dans les DL, pas de distinction entre
  - *catégories* = propriétés rigides (`Femelle`, `Mâle`)
  - *rôles* (dans le sens ontologique) = propriétés anti-rigides (`Mère`, `Enfant`, `Riche`)

  ⇒ ontologie = ‘photo’ à un instant temporel donné

Introduction : avantages des DL

- sémantique claire et bien définie
  - fragment de la logique du premier ordre
- tâches de raisonnement décidables
  - algorithmes optimaux
  - plus difficiles que pour la logique propositionnelle (les DL sont plus expressifs)
- applications multiples
  - web sémantique
  - génie du logiciel (diagrammes UML exprimables en DL)
  - bases de données (modèles entité-relation exprimables en DL)
- flexible
  - hiérarchie de DL
  - expressivité $\sim$ complexité

Introduction : l'aspect raisonnement

raisonner = 'extraire de la KB plus que ce qu'il y a écrit'

- exemple :

$$\mathcal{A} = \{\mathtt{pèreDe(Bob,Charles)}, \mathtt{pèreDe(Charles,Dan)}\}$$

$$\mathcal{T} = \{\exists \mathtt{pèreDe}.\exists \mathtt{pèreDe}.\top \sqsubseteq \exists \mathtt{GrandPèreDe}.\top\}$$

  - KB ne contient pas GrandPèreDe(Bob,Dan)
  - KB a comme conséquence logique GrandPèreDe(Bob,Dan) !
- services de raisonnement :
  - inférence de propriétés d'individus
  - inférence de relations entre individus
  - subsomption de concepts
  - classification = calculer la hierarchie de subsomption entre concepts
  - consistance
  - non-vacuité de concepts (pas de A tel que $KB \models A \equiv \bot$)
  - non-redondance de concepts (pas de A, A' t.q. $KB \models A \equiv A'$)

Introduction : l'aspect raisonnement

raisonner = 'extraire de la KB plus que ce qu'il y a écrit'

- exemple :

$$\mathcal{A} = \{\mathtt{p\grave{e}reDe(Bob,Charles)}, \mathtt{p\grave{e}reDe(Charles,Dan)}\}$$

$$\mathcal{T} = \{\exists\mathtt{p\grave{e}reDe}.\exists\mathtt{p\grave{e}reDe}.\top \sqsubseteq \exists\mathtt{GrandP\grave{e}reDe}.\top\}$$

  - KB ne contient pas GrandPèreDe(Bob,Dan)
  - KB a comme conséquence logique GrandPèreDe(Bob,Dan) !
- services de raisonnement :
  - inférence de propriétés d'individus
  - inférence de relations entre individus
  - subsomption de concepts
  - classification = calculer la hierarchie de subsomption entre concepts
  - consistance
  - non-vacuité de concepts (pas de A tel que $KB \models \mathtt{A} \equiv \bot$)
  - non-redondance de concepts (pas de A, A' t.q. $KB \models \mathtt{A} \equiv \mathtt{A'}$)

Introduction : complexité vs. expressivité

- quelle expressivité du langage ?
  - quels constructeurs de concepts ?
    - pas de négation '$\neg$' ; pas de disjonction '$\sqcup$' ; ...
  - quels constructeurs de rôles ?
    - pas d'inclusion de rôles '$\sqsubseteq$' ; pas de composition de rôles '$\circ$' ; ...
  - quels restrictions de langage ?
    - dans $\exists R.C$ il faut que $C = \top$; ...
- expressivité du langage $\Rightarrow$ complexité du raisonnement:
  - PTIME = temps polynomial
  - NPTIME = temps polynomial non-détérministe
  - PSPACE = espace polynomial (PSPACE = NPSPACE)
  - EXPTIME = temps exponentiel
  - NEXPTIME = temps exponentiel non-détérministe
  - EXPSPACE = espace exponentiel

Outline

Langage

- ALC = Attributive Concept Language with Complements [Schmidt-Schauß & Smolka, 1991]
- convention :
  - $\mathtt{A}$, $\mathtt{A}_i$, ...: concepts atomiques
  - $\mathtt{C}$, $\mathtt{D}$, $\mathtt{C}_i$, ...: concepts arbitraires
  - $\mathtt{R}$, $\mathtt{S}$, $\mathtt{R}_i$, ...: rôles atomiques
  - pas de rôles complexes
- grammaire des concepts (‘BNF’) :

$$\mathtt{C} ::= \mathtt{A} \mid \top \mid \bot \mid \neg \mathtt{C} \mid \mathtt{C} \sqcap \mathtt{C} \mid \mathtt{C} \sqcup \mathtt{C} \mid \forall \mathtt{R}.\mathtt{C} \mid \exists \mathtt{R}.\mathtt{C}$$

où $\mathtt{A}$ est un concept atomique et $\mathtt{R}$ est un rôle atomique

  - $\mathtt{A}$, $\mathtt{R}$: symboles terminaux
  - $\mathtt{C}$: symbole non-terminal

Langage : lecture des concepts complexes

$\top$	'tout'	concept universel
$\bot$	'rien'	concept vide
$\neg C$	'non C'	complément (ou : négation)
$C \sqcap D$	'C et D'	intersection (ou : conjonction)
$C \sqcup D$	'C ou D'	union (ou : disjonction)
$\forall R.C$	'tous les R-successeurs sont dans C'	quantification universelle restreinte
$\exists R.C$	'il existe un R-successeur qui est dans C'	quantification existentielle restreinte

Langage : exemples

- concepts atomiques :
  - $\mathtt{Personne}$, $\mathtt{Masculin}$, $\mathtt{Féminine}$, $\mathtt{Mère}$, $\mathtt{Enfant}$, $\mathtt{Riche}$
- rôles atomiques :
  - $\mathtt{parentDe}$, $\mathtt{mèreDe}$, $\mathtt{pèreDe}$
- concepts complexes :
  - $\mathtt{Personne} \sqcap \mathtt{Féminine}$
  - $\mathtt{Personne} \sqcap \neg\mathtt{Féminine}$
  - $\mathtt{Personne} \sqcap \exists\mathtt{parentDe}.\top$
  - $\mathtt{Personne} \sqcap \forall\mathtt{parentDe}.\bot$
  - $\mathtt{Personne} \sqcap \exists\mathtt{parentDe}.\top \sqcap \forall\mathtt{parentDe}.\mathtt{Féminine}$
  - $\mathtt{Personne} \sqcap (\mathtt{Riche} \sqcup \exists\mathtt{parentDe}.\mathtt{Riche})$
  - $\mathtt{Personne} \sqcap \exists\mathtt{parentDe}.\exists\mathtt{parentDe}.\top$

Langage : exercices

- définir les concepts suivants à partir des concepts atomiques `Personne`, `Masculin`, `Féminine`, `Riche` et des rôles atomiques `parentDe`, `sœurDe` et `frèreDe` :
  1. être une femme
  2. être une mère
  3. être une mère qui n'a que des garçons
  4. être un oncle
  5. être un grand-oncle
  6. être un quelqu'un qui a un neveu riche
  7. être quelqu'un qui a un oncle riche ??
     $\Rightarrow$ pas de converse dans ALC
  8. être quelqu'un qui n'a qu'un seul enfant ??
     $\Rightarrow$ pas de restriction de cardinalité dans ALC

Sémantique

interprétation $\mathcal{I} = (\Delta^{\mathcal{I}}, .^{\mathcal{I}})$ telle que :

- $\Delta^{\mathcal{I}}$ est un ensemble non-vide (domaine)
- $.^{\mathcal{I}}$: *ConceptsAtomiques* $\longrightarrow 2^{\Delta^{\mathcal{I}}}$ (interprétation des concepts)
- $.^{\mathcal{I}}$: *RolesAtomiques* $\longrightarrow 2^{\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}}$ (interprétation des rôles)

- une interprétation donne du sens aux concepts et rôles atomiques :
  - $\mathrm{C}^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ (concept = prédicat unaire)
  - $\mathrm{R}^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ (rôle = prédicat binaire)
- différence avec la logique du premier ordre :
  - pas d'interprétation de prédicats *n*-aires pour $n \geq 3$

Sémantique

interprétation $\mathcal{I} = (\Delta^{\mathcal{I}}, .^{\mathcal{I}})$ telle que :

- $\Delta^{\mathcal{I}}$ est un ensemble non-vide (domaine)
- $.^{\mathcal{I}}$: *ConceptsAtomiques* $\longrightarrow 2^{\Delta^{\mathcal{I}}}$ (interprétation des concepts)
- $.^{\mathcal{I}}$: *RolesAtomiques* $\longrightarrow 2^{\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}}$ (interprétation des rôles)

- une interprétation donne du sens aux concepts et rôles atomiques :
  - $\mathtt{C}^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ (concept = prédicat unaire)
  - $\mathtt{R}^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ (rôle = prédicat binaire)
- différence avec la logique du premier ordre :
  - pas d'interprétation de prédicats *n*-aires pour $n \geq 3$

Sémantique (suite)

interprétation des concepts complexes :

$$
\begin{aligned}
\top^{\mathcal{I}} &= \Delta^{\mathcal{I}} \\
\bot^{\mathcal{I}} &= \emptyset \\
(\neg \mathtt{C})^{\mathcal{I}} &= \Delta^{\mathcal{I}} \setminus \mathtt{C}^{\mathcal{I}} \\
(\mathtt{C} \sqcap \mathtt{D})^{\mathcal{I}} &= \mathtt{C}^{\mathcal{I}} \cap \mathtt{D}^{\mathcal{I}} \\
(\mathtt{C} \sqcup \mathtt{D})^{\mathcal{I}} &= \mathtt{C}^{\mathcal{I}} \cup \mathtt{D}^{\mathcal{I}} \\
(\forall \mathtt{R}.\mathtt{C})^{\mathcal{I}} &= \left\{a \in \Delta^{\mathcal{I}} \;:\; \text{pour tout } b \in \Delta^{\mathcal{I}}, \text{si } (a,b) \in \mathtt{R}^{\mathcal{I}} \text{ alors } b \in \mathtt{C}^{\mathcal{I}}\right\} \\
(\exists \mathtt{R}.\mathtt{C})^{\mathcal{I}} &= \left\{a \in \Delta^{\mathcal{I}} \;:\; \text{il existe } b \in \Delta^{\mathcal{I}} \text{ tel que } (a,b) \in \mathtt{R}^{\mathcal{I}} \text{ et } b \in \mathtt{C}^{\mathcal{I}}\right\}
\end{aligned}
$$

Sémantique : quelques équivalences

- équivalences sémantiques :

$$(\neg\neg C)^{\mathcal{I}} = C^{\mathcal{I}}$$

$$(\neg(C \sqcap D))^{\mathcal{I}} = \left(\neg C^{\mathcal{I}} \sqcup \neg D\right)^{\mathcal{I}}$$

$$(\neg(C \sqcup D))^{\mathcal{I}} = \left(\neg C^{\mathcal{I}} \sqcap \neg D\right)^{\mathcal{I}}$$

$$(\neg\forall R.C)^{\mathcal{I}} = (\exists R.\neg C)^{\mathcal{I}}$$

$$(\neg\exists R.C)^{\mathcal{I}} = (\forall R.\neg C)^{\mathcal{I}}$$

Proposition (forme normale négative)

Tout concept C peut être transformé en un concept $nnf(C)$ tel que

1. *dans $nnf(C)$, la négation apparaît seulement devant des concepts atomiques,*
2. *pour toute interprétation $\mathcal{I}$, $(nnf(C))^{\mathcal{I}} = C^{\mathcal{I}}$.*

- exemple : $\neg\forall \mathtt{parentDe}.\bot$ (... il manque une équivalence)

Sémantique : quelques équivalences

- équivalences sémantiques :

$$
\begin{aligned}
(\neg\neg \mathtt{C})^{\mathcal{I}} &= C^{\mathcal{I}} \\
(\neg(\mathtt{C} \sqcap \mathtt{D}))^{\mathcal{I}} &= \left(\neg \mathtt{C}^{\mathcal{I}} \sqcup \neg \mathtt{D}\right)^{\mathcal{I}} \\
(\neg(\mathtt{C} \sqcup \mathtt{D}))^{\mathcal{I}} &= \left(\neg \mathtt{C}^{\mathcal{I}} \sqcap \neg \mathtt{D}\right)^{\mathcal{I}} \\
(\neg\forall \mathtt{R}.\mathtt{C})^{\mathcal{I}} &= (\exists \mathtt{R}.\neg \mathtt{C})^{\mathcal{I}} \\
(\neg\exists \mathtt{R}.\mathtt{C})^{\mathcal{I}} &= (\forall \mathtt{R}.\neg \mathtt{C})^{\mathcal{I}}
\end{aligned}
$$

Proposition (forme normale négative)

Tout concept C peut être transformé en un concept $nnf(C)$ tel que

1. *dans $nnf(C)$, la négation apparaît seulement devant des concepts atomiques,*
2. *pour toute interprétation $\mathcal{I}$, $(nnf(C))^{\mathcal{I}} = C^{\mathcal{I}}$.*

- exemple : $\neg\forall \mathtt{parentDe}.\bot$ (. . . il manque une équivalence)

Sémantique : quelques équivalences

- équivalences sémantiques :

$$(\neg\neg \mathtt{C})^{\mathcal{I}} = C^{\mathcal{I}}$$
$$(\neg(\mathtt{C} \sqcap \mathtt{D}))^{\mathcal{I}} = \left(\neg \mathtt{C}^{\mathcal{I}} \sqcup \neg \mathtt{D}\right)^{\mathcal{I}}$$
$$(\neg(\mathtt{C} \sqcup \mathtt{D}))^{\mathcal{I}} = \left(\neg \mathtt{C}^{\mathcal{I}} \sqcap \neg \mathtt{D}\right)^{\mathcal{I}}$$
$$(\neg \forall \mathtt{R}.\mathtt{C})^{\mathcal{I}} = (\exists \mathtt{R}.\neg \mathtt{C})^{\mathcal{I}}$$
$$(\neg \exists \mathtt{R}.\mathtt{C})^{\mathcal{I}} = (\forall \mathtt{R}.\neg \mathtt{C})^{\mathcal{I}}$$

Proposition (forme normale négative)

Tout concept C peut être transformé en un concept nnf(C) tel que

1. *dans nnf(C), la négation apparaît seulement devant des concepts atomiques,*
2. *pour toute interprétation $\mathcal{I}$, $(nnf(C))^{\mathcal{I}} = C^{\mathcal{I}}$.*

- exemple : $\neg \forall \mathtt{parentDe}.\bot$ (... il manque une équivalence)

Outline

Le langage ALCN : ALC plus restrictions de cardinalité

- ALC plus un nouveau constructeurs de concept : restriction de cardinalité
- grammaire des concepts (‘BNF’) :

$$\mathtt{C} ::= \mathtt{A} \mid \top \mid \bot \mid \neg\mathtt{C} \mid \mathtt{C} \sqcap \mathtt{C} \mid \mathtt{C} \sqcup \mathtt{C} \mid \forall\mathtt{R}.\mathtt{C} \mid \exists\mathtt{R}.\mathtt{C} \mid \leq n\ \mathtt{R} \mid \geq n\ \mathtt{R}$$

  où $\mathtt{A}$ est un concept atomique et $\mathtt{R}$ est un rôle atomique et $n \geq 0$

- lecture :
  - $\leq n\ \mathtt{R}$ = ‘il y a au plus n $\mathtt{R}$-successeurs’
  - $\geq n\ \mathtt{R}$ = ‘il y a au moins n $\mathtt{R}$-successeurs’
- exercice :
  - définir le concept $=n\ \mathtt{R}$ (‘il y a exactement n $\mathtt{R}$-successeurs’)

Le langage ALCN : ALC plus restrictions de cardinalité

- ALC plus un nouveau constructeurs de concept : restriction de cardinalité
- grammaire des concepts ('BNF') :

$$\mathtt{C} ::= \mathtt{A} \mid \top \mid \bot \mid \neg\mathtt{C} \mid \mathtt{C} \sqcap \mathtt{C} \mid \mathtt{C} \sqcup \mathtt{C} \mid \forall\mathtt{R}.\mathtt{C} \mid \exists\mathtt{R}.\mathtt{C} \mid \leq n\ \mathtt{R} \mid \geq n\ \mathtt{R}$$

où $\mathtt{A}$ est un concept atomique et $\mathtt{R}$ est un rôle atomique et $n \geq 0$

- lecture :
  - $\leq n\ \mathtt{R}$ = 'il y a au plus n $\mathtt{R}$-successeurs'
  - $\geq n\ \mathtt{R}$ = 'il y a au moins n $\mathtt{R}$-successeurs'
- exercice :
  - définir le concept $=n\ \mathtt{R}$ ('il y a exactement n $\mathtt{R}$-successeurs')

ALCN : sémantique

- les mêmes interprétations que dans ALC :
  - $\mathcal{I} = (\Delta^\mathcal{I}, .^\mathcal{I})$ telle que
    - $\Delta^\mathcal{I}$ est un ensemble non-vide (domaine)
    - $.^\mathcal{I}$: *ConceptsAtomiques* $\longrightarrow 2^{\Delta^\mathcal{I}}$ (interprétation des concepts)
    - $.^\mathcal{I}$: *RolesAtomiques* $\longrightarrow 2^{\Delta^\mathcal{I} \times \Delta^\mathcal{I}}$ (interprétation des rôles)
- interprétation des concepts complexes :

$$(\leq n\ \mathrm{R})^\mathcal{I} = \left\{a \in \Delta^\mathcal{I} \ : \ Card(\{b \in \Delta \ : \ (a,b) \in \mathrm{R}^\mathcal{I}\}) \leq n\right\}$$

$$(\geq n\ \mathrm{R})^\mathcal{I} = \ldots$$

ALCN : exercices

- décrire les concepts suivants :
  1. être une personne avec au moins trois enfants
  2. être une personne avec exactement trois enfants
  3. être un cours avec au plus 15 participants dont tous sont des étudiants de master
     - concepts primitifs : `Cours`, `ÉtudiantEnMaster`
     - rôle primitif : `aParticipant`
  4. être une grand-mère avec une fille qui a exactement deux fils
- étendre la forme normale négative aux restrictions de cardinalité

Au-delà de ALCN : restrictions de cardinalité *qualifiées*

- grammaire des concepts ('BNF') :

$$\mathtt{C} ::= \mathtt{A} \mid \top \mid \bot \mid \neg\mathtt{C} \mid \mathtt{C} \sqcap \mathtt{C} \mid \mathtt{C} \sqcup \mathtt{C} \mid \leq n\, \mathtt{R.C} \mid \geq n\, \mathtt{R.C}$$

  où $\mathtt{A}$ est un concept atomique, $\mathtt{R}$ est un rôle atomique et $n \geq 0$
- lecture :
  - $\leq n\, \mathtt{R.C}$ = 'au plus n $\mathtt{R}$-successeurs sont dans $\mathtt{C}$'
  - $\geq n\, \mathtt{R.C}$ = 'au moins n $\mathtt{R}$-successeurs sont dans $\mathtt{C}$'

Au-delà de ALCN : restrictions de cardinalité *qualifiées*

- interprétation :

$$(\leq n\ \mathrm{R.C})^{\mathcal{I}} = \left\{ a \in \Delta^{\mathcal{I}} : Card(\{b \in \Delta : (a,b) \in \mathrm{R}^{\mathcal{I}} \ \underline{\text{et } b \in \mathrm{C}^{\mathcal{I}}}\}) \leq n \right\}$$

$$(\geq n\ \mathrm{R.C})^{\mathcal{I}} = \ldots$$

- au moins aussi expressif que ALCN :
  - $\leq n\ \mathrm{R} \stackrel{\text{def}}{=} \ldots$
  - $\geq n\ \mathrm{R} \stackrel{\text{def}}{=} \ldots$
  - $\exists \mathrm{R.C} \stackrel{\text{def}}{=} \ldots$
  - $\forall \mathrm{R.C} \stackrel{\text{def}}{=} \ldots$
- strictement plus expressif que ALCN :
  - les restrictions de cardinalité qualifiées ne peuvent pas toutes être exprimées dans ALCN

Outline

Traduction de ALCN en logique du premier ordre

$$\boxed{\text{concept } \mathtt{C} \longmapsto \text{formule } \Phi(\mathtt{C}, x) \text{ de FOL}}$$

- $\Phi(\mathtt{C}, x)$ a une seule variable libre : x ('l'individu actuel')

$$\begin{aligned}
\Phi(\mathtt{A}, x) &= \mathtt{A}(x) \\
\Phi(\top, x) &= \top \\
\Phi(\bot, x) &= \bot \\
\Phi(\neg\mathtt{C}, x) &= \neg\Phi(\mathtt{C}, x) \\
\Phi(\mathtt{C} \sqcap \mathtt{D}, x) &= \Phi(\mathtt{C}, x) \wedge \Phi(\mathtt{D}, x) \\
\Phi(\mathtt{C} \sqcup \mathtt{D}, x) &= \Phi(\mathtt{C}, x) \vee \Phi(\mathtt{D}, x) \\
\Phi(\forall\mathtt{R}.\mathtt{C}, x) &= \forall y\,(\mathtt{R}(x, y) \rightarrow \Phi(\mathtt{C}, y)) \\
\Phi(\exists\mathtt{R}.\mathtt{C}, x) &= \ldots \\
\Phi(\geq 2\ \mathtt{R}, x) &= \ldots \\
\Phi(\geq n\ \mathtt{R}, x) &= \ldots \\
\Phi(\leq n\ \mathtt{R}, x) &= \ldots
\end{aligned}$$

Traduction de ALCN en logique du premier ordre (suite)

$$\Phi(\forall \mathrm{R}.\mathrm{C}, x) = \forall y \, (\mathrm{R}(x, y) \rightarrow \Phi(\mathrm{C}, y))$$

$$\Phi(\exists \mathrm{R}.\mathrm{C}, x) = \exists y \, (\mathrm{R}(x, y) \wedge \Phi(\mathrm{C}, y))$$

$$\Phi(\leq n \, \mathrm{R}, x) = \forall y_1 \ldots \forall y_{n+1} \left((\mathrm{R}(x, y_1) \wedge \ldots \wedge \mathrm{R}(x, y_{n+1})) \rightarrow \bigvee_{i<j\leq n+1} y_i = y_j \right)$$

$$\Phi(\geq n \, \mathrm{R}, x) = \exists y_1 \ldots \exists y_n \left(\mathrm{R}(x, y_1) \wedge \ldots \wedge \mathrm{R}(x, y_n) \wedge \bigwedge_{i<j\leq n} y_i \neq y_j \right)$$

- exercice : traduire en FOL
  1. $\mathtt{Personne} \sqcap \exists \mathtt{parentDe}.\exists \mathtt{parentDe}.\mathtt{Mâle}$
  2. $\mathtt{Personne} \sqcap \leq 2 \; \mathtt{parentDe}$

Traduction de ALCN en logique du premier ordre (suite)

Proposition

Pour tout concept $\mathtt{C}$ *et pour toute interprétation* $(\Delta^{\mathcal{I}}, (.)^{\mathcal{I}})$ *:*

$$C^{\mathcal{I}} = \left\{ a \in \Delta^{\mathcal{I}} \;:\; \mathcal{I} \Vdash_{FOL} \Phi(\mathtt{C}, a) \right\}$$

démonstration par induction sur la forme de $\mathtt{C}$:

pour tout $\mathcal{I}$ et tout $a \in \Delta^{\mathcal{I}}$, $a \in C^{\mathcal{I}}$ ssi $\mathcal{I} \Vdash_{FOL} \Phi(\mathtt{C}, a)$

1. si $\mathtt{C}$ est un concept atomique $\mathtt{A}$ alors :
   $a \in \mathtt{A}^{\mathcal{I}}$ ssi $\mathcal{I} \Vdash_{FOL} \mathtt{A}(a)$
   ssi $\mathcal{I} \Vdash_{FOL} \Phi(\mathtt{A}, a)$
2. si $\mathtt{C}$ est de la forme $\top$ alors ...
3. si $\mathtt{C}$ est de la forme $\bot$ alors ...
4. si $\mathtt{C}$ est de la forme $\neg\mathtt{D}$ alors :
   $a \in (\neg\mathtt{D})^{\mathcal{I}}$ ssi $a \notin \mathtt{D}^{\mathcal{I}}$
   ssi $\mathcal{I} \nVdash_{FOL} \Phi(\mathtt{D}, a)$ (par H.I.)
   ssi $\mathcal{I} \Vdash_{FOL} \Phi(\neg\mathtt{D}, a)$
5. si ...

Traduction de ALCN en logique du premier ordre (suite)

Proposition

Pour tout concept $\mathtt{C}$ *et pour toute interprétation* $(\Delta^{\mathcal{I}}, (.)^{\mathcal{I}})$ *:*

$$C^{\mathcal{I}} = \left\{ a \in \Delta^{\mathcal{I}} \ : \ \mathcal{I} \Vdash_{FOL} \Phi(\mathtt{C}, a) \right\}$$

démonstration par induction sur la forme de $\mathtt{C}$:

pour tout $\mathcal{I}$ et tout $a \in \Delta^{\mathcal{I}}$, $a \in C^{\mathcal{I}}$ ssi $\mathcal{I} \Vdash_{FOL} \Phi(\mathtt{C}, a)$

1. si $\mathtt{C}$ est un concept atomique $\mathtt{A}$ alors :
   $a \in \mathtt{A}^{\mathcal{I}}$ ssi $\mathcal{I} \Vdash_{FOL} \mathtt{A}(a)$
   ssi $\mathcal{I} \Vdash_{FOL} \Phi(\mathtt{A}, a)$
2. si $\mathtt{C}$ est de la forme $\top$ alors ...
3. si $\mathtt{C}$ est de la forme $\bot$ alors ...
4. si $\mathtt{C}$ est de la forme $\neg\mathtt{D}$ alors :
   $a \in (\neg\mathtt{D})^{\mathcal{I}}$ ssi $a \notin \mathtt{D}^{\mathcal{I}}$
   ssi $\mathcal{I} \nVdash_{FOL} \Phi(\mathtt{D}, a)$ (par H.I.)
   ssi $\mathcal{I} \Vdash_{FOL} \Phi(\neg\mathtt{D}, a)$
5. si ...

Outline

La logique modale en un transparent

- logique multimodale K_n :

$\Box_{\mathtt{R}}\varphi$ = 'φ est vrai dans *tous* les états accessibles via $\mathtt{R}$'
= 'φ est vrai après tout exécution du programme $\mathtt{R}$'
= 'φ est nécessaire'
= ...

$\Box_{\mathtt{R}}\varphi$ = 'φ est vrai dans *quelques* les états accessibles via $\mathtt{R}$'
= 'il existe une exécution du programme $\mathtt{R}$ après laquelle φ est vrai'
= 'φ est possible'
= ...

Traduction de ALCN en logique modale

- fonction de traduction :

$$
\begin{aligned}
\mu(\mathtt{A}) &= \mathtt{A} && \text{si } \mathtt{A} \text{ est un concept atomique}\\
\mu(\top) &= \top\\
\mu(\bot) &= \bot\\
\mu(\neg \mathtt{C}) &= \neg\mu(\mathtt{C})\\
\mu(\mathtt{C} \sqcap \mathtt{D}) &= \mu(\mathtt{C}) \wedge \mu(\mathtt{D})\\
\mu(\mathtt{C} \sqcup \mathtt{D}) &= \mu(\mathtt{C}) \vee \mu(\mathtt{D})\\
\mu(\forall \mathrm{R}.\mathtt{C}) &= \Box_{\mathrm{R}}\mu(\mathtt{C})\\
\mu(\exists \mathrm{R}.\mathtt{C}) &= \Diamond_{\mathrm{R}}\mu(\mathtt{C})
\end{aligned}
$$

Outline

TBox générale

- $\mathtt{C} \sqsubseteq \mathtt{D}$: 'axiome général d'inclusion de concept'

$$\mathcal{I} \Vdash \mathtt{C} \sqsubseteq \mathtt{D} \text{ ssi } \mathtt{C}^{\mathcal{I}} \subseteq \mathtt{D}^{\mathcal{I}}$$

- $\mathtt{C} \equiv \mathtt{D}$: 'axiome général d'équivalence de concept'

$$\mathcal{I} \Vdash \mathtt{C} \equiv \mathtt{D} \text{ ssi } \mathtt{C}^{\mathcal{I}} = \mathtt{D}^{\mathcal{I}}$$

- $\mathcal{T}$ = ensemble fini d'axiomes généraux (d'inclusion ou d'équivalence)
  = TBox générale

$$\mathcal{I} \Vdash \mathcal{T} \text{ ssi } \mathcal{I} \Vdash t \text{ pour tout } t \in \mathcal{T}$$

- élimination des axiomes d'inclusion :
  - remplacer $\mathtt{C} \sqsubseteq \mathtt{D}$ par $\mathtt{C} \equiv \mathtt{D} \sqcap \mathtt{D}'$, où D' est nouveau
  - équivalent à la TBox d'origine *en ce qui concerne le langage de $\mathcal{T}$*

TBox générale

- $\mathtt{C} \sqsubseteq \mathtt{D}$: 'axiome général d'inclusion de concept'

$$\mathcal{I} \Vdash \mathtt{C} \sqsubseteq \mathtt{D} \text{ ssi } \mathtt{C}^{\mathcal{I}} \subseteq \mathtt{D}^{\mathcal{I}}$$

- $\mathtt{C} \equiv \mathtt{D}$: 'axiome général d'équivalence de concept'

$$\mathcal{I} \Vdash \mathtt{C} \equiv \mathtt{D} \text{ ssi } \mathtt{C}^{\mathcal{I}} = \mathtt{D}^{\mathcal{I}}$$

- $\mathcal{T}$ = ensemble fini d'axiomes généraux (d'inclusion ou d'équivalence)
  = TBox générale

$$\mathcal{I} \Vdash \mathcal{T} \text{ ssi } \mathcal{I} \Vdash t \text{ pour tout } t \in \mathcal{T}$$

- élimination des axiomes d'inclusion :
  - remplacer $\mathtt{C} \sqsubseteq \mathtt{D}$ par $\mathtt{C} \equiv \mathtt{D} \sqcap \mathtt{D}'$, où $\mathtt{D}$' est nouveau
  - équivalent à la TBox d'origine *en ce qui concerne le langage de $\mathcal{T}$*

TBox acyclique

- $\mathtt{C} \equiv \mathtt{D}$ est une définition ssi $\mathtt{C}$ est atomique
  - $\mathtt{A} \equiv \mathtt{D}$ = 'le nom $\mathtt{A}$ est défini par $\mathtt{D}$'
- une TBox générale $\mathcal{T}$ est une TBox acyclique ssi
  1. $\mathcal{T}$ ne contient que des définitions
  2. pas de définitions multiples
     - interdit :
       $\mathtt{A} \equiv \mathtt{C}, \mathtt{A} \equiv \mathtt{D} \in \mathcal{T}$ et $\mathtt{C} \neq \mathtt{C}'$
  3. pas de cycles
     - interdit :
       $\mathtt{A}_1 \equiv \ldots \mathtt{A}_2 \ldots$
       $\mathtt{A}_2 \equiv \ldots \mathtt{A}_3 \ldots$
       $\ldots$
       $\mathtt{A}_n \equiv \ldots \mathtt{A}_1 \ldots$
- concepts atomiques primitifs vs. concepts atomiques définis

$\Rightarrow$ acyclicité intéressant pour le calcul

Outline

ABox = *assertion box*

- $a, b, \ldots$: noms d'individus
  - $\mathtt{Alice}, \mathtt{Bob}, \mathtt{Charles}$
- $\mathtt{C}(a)$: instance de concept
  - $(\mathtt{Mâle}) \sqcap \mathtt{Personne})(\mathtt{Charles})$
- $\mathtt{R}(a, b)$: instance de rôle
  - $\mathtt{parentDe(Alice,Charles)}$
- ABox $\mathcal{A}$ = ensemble fini d'instances de concepts et rôles
  - $\mathcal{I} \Vdash \mathcal{A}$ ssi
    1. si $\mathtt{C}(a) \in \mathcal{A}$ alors $a^{\mathcal{I}} \in \mathtt{C}^{\mathcal{I}}$
    2. si $\mathtt{R}(a, b) \in \mathcal{A}$ alors $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in \mathtt{R}^{\mathcal{I}}$
    3. si $a \neq b$ alors $a^{\mathcal{I}} \neq b^{\mathcal{I}}$
       'hypothèse de nom unique' ; exemple : $\mathtt{Bob}^{\mathcal{I}} \neq \mathtt{Charles}^{\mathcal{I}}$

Exemple d'une KB généalogique

$$\mathcal{T}_{gen} = \{ \begin{array}{rcl} \mathtt{Femme} & \equiv & \mathtt{Personne} \sqcap \mathtt{Femelle}\,, \\ \mathtt{Homme} & \equiv & \mathtt{Personne} \sqcap \mathtt{Mâle}\,, \\ \mathtt{Mère} & \equiv & \mathtt{Femme} \sqcap \exists \mathtt{parentDe.Personne}\,, \\ \mathtt{Père} & \equiv & \mathtt{Homme} \sqcap \exists \mathtt{parentDe.Personne}\,, \\ \mathtt{Parent} & \equiv & \mathtt{Mère} \sqcup \mathtt{Père}\,, \\ \mathtt{MèreSansFille} & \equiv & \mathtt{Mère} \sqcap \forall \mathtt{parentDe.}\neg\mathtt{Femme} \end{array} \}$$

$$\mathcal{A}_{gen} = \{ \begin{array}{l} \mathtt{Femme(Alice)}, \\ \mathtt{Homme(Bob)}, \\ \mathtt{parentDe(Alice,Charles)}, \\ \mathtt{parentDe(Bob,Charles)}, \\ \ldots \end{array} \}$$

Outline

Tâches de raisonnement

$\boxed{\text{C est satisfaisable ssi il existe une interprétation } \mathcal{I} \text{ tel que } C^{\mathcal{I}} \neq \emptyset}$

- subsomption :
  - $C \sqsubseteq D$ ssi $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ pour tout interprétation $\mathcal{I}$
  - réductible à un test de consistance
    - $C \sqsubseteq D$ ssi $C \sqcap \neg D$ est insatisfaisable
- équivalence :
  - C et D sont équivalents ssi $C^{\mathcal{I}} = D^{\mathcal{I}}$ pour tout interprétation $\mathcal{I}$
  - réductible à deux tests de subsomption
- exclusivité (*disjointness*) :
  - C et D sont disjoints ssi $C^{\mathcal{I}} \cap D^{\mathcal{I}} = \emptyset$ pour tout interprétation $\mathcal{I}$
  - réductible à un test de consistance
    - C et D sont disjoints ssi $C \sqcap D$ est insatisfaisable

Tâches de raisonnement

$\mathtt{C}$ est satisfaisable ssi il existe une interprétation $\mathcal{I}$ tel que $\mathtt{C}^{\mathcal{I}} \neq \emptyset$

- subsomption :
  - $\mathtt{C} \sqsubseteq \mathtt{D}$ ssi $\mathtt{C}^{\mathcal{I}} \subseteq \mathtt{D}^{\mathcal{I}}$ pour tout interprétation $\mathcal{I}$
  - réductible à un test de consistance
    - $\mathtt{C} \sqsubseteq \mathtt{D}$ ssi $\mathtt{C} \sqcap \neg\mathtt{D}$ est insatisfaisable
- équivalence :
  - $\mathtt{C}$ et $\mathtt{D}$ sont équivalents ssi $\mathtt{C}^{\mathcal{I}} = \mathtt{D}^{\mathcal{I}}$ pour tout interprétation $\mathcal{I}$
  - réductible à deux tests de subsomption
- exclusivité (*disjointness*) :
  - $\mathtt{C}$ et $\mathtt{D}$ sont disjoints ssi $\mathtt{C}^{\mathcal{I}} \cap \mathtt{D}^{\mathcal{I}} = \emptyset$ pour tout interprétation $\mathcal{I}$
  - réductible à un test de consistance
    - $\mathtt{C}$ et $\mathtt{D}$ sont disjoints ssi $\mathtt{C} \sqcap \mathtt{D}$ est insatisfaisable

Tâches de raisonnement p.r.à une TBox

C est satisfaisable p.r.à $\mathcal{T}$ ssi il existe $\mathcal{I}$ tel que $\mathcal{I} \Vdash \mathcal{T}$ et $C^{\mathcal{I}} \neq \emptyset$

- subsomption p.r.à $\mathcal{T}$
- équivalence p.r.à $\mathcal{T}$
- exclusivité (*disjointness*) p.r.à $\mathcal{T}$
- classification = calculer toute hiérarchie de subsomption d'une TBox (graphe)

$\Rightarrow$ tout se réduit à des tests de satisfaisabilité p.r.à $\mathcal{T}$

$\Rightarrow$ et si la TBox $\mathcal{T}$ est acyclique alors on peut faire encore plus simple . . .

Tâches de raisonnement p.r.à une TBox

$\boxed{\text{C est satisfaisable p.r.à } \mathcal{T} \text{ ssi il existe } \mathcal{I} \text{ tel que } \mathcal{I} \Vdash \mathcal{T} \text{ et } \mathsf{C}^{\mathcal{I}} \neq \emptyset}$

- subsomption p.r.à $\mathcal{T}$
- équivalence p.r.à $\mathcal{T}$
- exclusivité (*disjointness*) p.r.à $\mathcal{T}$
- classification = calculer toute hiérarchie de subsomption d'une TBox (graphe)

$\Rightarrow$ tout se réduit à des tests de satisfaisabilité p.r.à $\mathcal{T}$

$\Rightarrow$ et si la TBox $\mathcal{T}$ est acyclique alors on peut faire encore plus simple ...

Comment se débarasser d'une TBox

- $\mathtt{C}^{\mathcal{T}}$ = expansion de $\mathtt{C}$ par $\mathcal{T}$:
  1. remplacer chaque concept non-primitif $\mathtt{A}_i$ dans $\mathtt{C}$ par la définition de $\mathtt{A}_i$ dans $\mathcal{T}$
  2. itérer jusqu'à ce qu'il n'y a plus de concepts non-primitifs (termine grâce à l'acyclique)
- $\mathcal{A}^{\mathcal{T}} = \{\mathtt{C}^{\mathcal{T}}(\mathtt{a}) \;:\; \mathtt{C}(\mathtt{a}) \in \mathcal{A}\} \cup \{\mathtt{R}(\mathtt{a},\mathtt{b}) \;:\; \mathtt{R}(\mathtt{a},\mathtt{b}) \in \mathcal{A}\}$
- exercice :

$$
\mathcal{T}_{gen} = \{ \begin{array}{rcl}
\mathtt{Femme} & \equiv & \mathtt{Personne} \sqcap \mathtt{Femelle}, \\
\mathtt{Homme} & \equiv & \mathtt{Personne} \sqcap \mathtt{Mâle}, \\
\mathtt{Mère} & \equiv & \mathtt{Femme} \sqcap \exists \mathtt{parentDe.Personne}, \\
\mathtt{Père} & \equiv & \mathtt{Homme} \sqcap \exists \mathtt{parentDe.Personne}, \\
\mathtt{Parent} & \equiv & \mathtt{Mère} \sqcup \mathtt{Père}, \\
\mathtt{MèreSansFille} & \equiv & \mathtt{Mère} \sqcap \forall \mathtt{parentDe.} \neg \mathtt{Femme}
\end{array} \}
$$

  1. identifier les concepts primitifs
  2. trouver les extensions de tous les concepts non-primitifs
  3. trouver l'extension de $\mathtt{Mère} \sqcap \forall \mathtt{parentDe.}\bot$

Comment se débarasser d'une TBox

- $\mathtt{C}^{\mathcal{T}}$ = expansion de $\mathtt{C}$ par $\mathcal{T}$:
  1. remplacer chaque concept non-primitif $\mathtt{A}_i$ dans $\mathtt{C}$ par la définition de $\mathtt{A}_i$ dans $\mathcal{T}$
  2. itérer jusqu'à ce qu'il n'y a plus de concepts non-primitifs (termine grâce à l'acyclique)
- $\mathcal{A}^{\mathcal{T}} = \{\mathtt{C}^{\mathcal{T}}(\mathtt{a}) : \mathtt{C}(\mathtt{a}) \in \mathcal{A}\} \cup \{\mathtt{R}(\mathtt{a},\mathtt{b}) : \mathtt{R}(\mathtt{a},\mathtt{b}) \in \mathcal{A}\}$
- exercice :

$$\mathcal{T}_{gen} = \{ \begin{array}{rcl} \mathtt{Femme} & \equiv & \mathtt{Personne} \sqcap \mathtt{Femelle}\ , \\ \mathtt{Homme} & \equiv & \mathtt{Personne} \sqcap \mathtt{Mâle}\ , \\ \mathtt{Mère} & \equiv & \mathtt{Femme} \sqcap \exists \mathtt{parentDe.Personne}\ , \\ \mathtt{Père} & \equiv & \mathtt{Homme} \sqcap \exists \mathtt{parentDe.Personne}\ , \\ \mathtt{Parent} & \equiv & \mathtt{Mère} \sqcup \mathtt{Père}\ , \\ \mathtt{MèreSansFille} & \equiv & \mathtt{Mère} \sqcap \forall \mathtt{parentDe.} \neg \mathtt{Femme} \end{array} \}$$

  1. identifier les concepts primitifs
  2. trouver les extensions de tous les concepts non-primitifs
  3. trouver l'extension de $\mathtt{Mère} \sqcap \forall \mathtt{parentDe.}\bot$

Comment se débarasser d'une TBox (suite)

Proposition

C est satisfaisable p.r.à $\mathcal{T}$ ssi $C^{\mathcal{T}}$ est satisfaisable.

- exemple : $\mathtt{Mère} \sqcap \forall \mathtt{parentDe}.\bot$ satisfaisable p.r.à $\mathcal{T}_{gen}$ ssi $(\mathtt{Mère} \sqcap \forall \mathtt{parentDe}.\bot)^{\mathcal{T}_{gen}} =$ $\mathtt{Personne} \sqcap \mathtt{Femelle} \sqcap \exists \mathtt{parentDe}.\mathtt{Personne} \sqcap \forall \mathtt{parentDe}.\bot$ est satisfaisable tout court
  ... ce qui n'est pas le cas
- exercice :
  - trouver $\mathcal{T}$ et $\mathtt{A}$ t.q. $\mathtt{A}^{\mathcal{T}}$ est exponentiellement plus long que $\mathcal{T}$

Tâches de raisonnement p.r.à une ABox

$\boxed{\text{une ABox } \mathcal{A} \text{ est satisfaisable ssi il existe } \mathcal{I} \text{ tel que } \mathcal{I} \Vdash \mathcal{A}}$

- satisfaisabilité p.r.à une TBox :
  - $\mathcal{A}$ satisfaisable p.r.à $\mathcal{T}$ ssi il existe $\mathcal{I}$ tel que $\mathcal{I} \Vdash \mathcal{T}$ et $\mathcal{I} \Vdash \mathcal{A}$
  - si $\mathcal{T}$ est acyclique : $\mathcal{A}$ satisfaisable p.r.à $\mathcal{T}$ ssi $\mathcal{A}^{\mathcal{T}}$ satisfaisable
  - subsume la satisfaisabilité d'un concept
    - C est satisfaisable ssi $\{\mathtt{C(a)}\}$ est satisfaisable, pour un individu *a* quelconque
- inférence de propriétés :
  - $\mathcal{T} \cup \mathcal{A} \models \mathtt{C(a)}$ ssi pour tout $\mathcal{I}$, si $\mathcal{I} \Vdash \mathcal{T}$ et $\mathcal{I} \Vdash \mathcal{A}$ alors $\mathcal{I} \Vdash \mathtt{C(a)}$
  - exemple :
    - $\mathcal{T}_{gen} \cup \{\mathtt{GrandMère(Alice)}\} \models \mathtt{Mère(Alice)}$
  - $\mathcal{T} \cup \mathcal{A} \models \mathtt{C(a)}$ ssi $\mathcal{A} \cup \{\neg\mathtt{C(a)}\}$ insatisfaisable p.r.à $\mathcal{T}$
- requête :
  - trouver tous les *a* tel que $\mathcal{T} \cup \mathcal{A} \models \mathtt{C}(a)$
- réalisation :
  - calculer les noms de concept les plus spécifiques de la TBox pour un individu donné (cf. la classification)
    - Bob est instance de Personne, Mâle, *Père*

Tâches de raisonnement p.r.à une ABox

une ABox $\mathcal{A}$ est satisfaisable ssi il existe $\mathcal{I}$ tel que $\mathcal{I} \Vdash \mathcal{A}$

- satisfaisabilité p.r.à une TBox :
  - $\mathcal{A}$ satisfaisable p.r.à $\mathcal{T}$ ssi il existe $\mathcal{I}$ tel que $\mathcal{I} \Vdash \mathcal{T}$ et $\mathcal{I} \Vdash \mathcal{A}$
  - si $\mathcal{T}$ est acyclique : $\mathcal{A}$ satisfaisable p.r.à $\mathcal{T}$ ssi $\mathcal{A}^{\mathcal{T}}$ satisfaisable
  - subsume la satisfaisabilité d'un concept
    - $\mathtt{C}$ est satisfaisable ssi $\{\mathtt{C(a)}\}$ est satisfaisable, pour un individu *a* quelconque
- inférence de propriétés :
  - $\mathcal{T} \cup \mathcal{A} \models \mathtt{C(a)}$ ssi pour tout $\mathcal{I}$, si $\mathcal{I} \Vdash \mathcal{T}$ et $\mathcal{I} \Vdash \mathcal{A}$ alors $\mathcal{I} \Vdash \mathtt{C(a)}$
  - exemple :
    - $\mathcal{T}_{gen} \cup \{\mathtt{GrandMère(Alice)}\} \models \mathtt{Mère(Alice)}$
  - $\mathcal{T} \cup \mathcal{A} \models \mathtt{C(a)}$ ssi $\mathcal{A} \cup \{\neg\mathtt{C(a)}\}$ insatisfaisable p.r.à $\mathcal{T}$
- requête :
  - trouver tous les *a* tel que $\mathcal{T} \cup \mathcal{A} \models \mathtt{C}(a)$
- réalisation :
  - calculer les noms de concept les plus spécifiques de la TBox pour un individu donné (cf. la classification)
    - `Bob` est instance de `Personne`, `Mâle`, *`Père`*

Outline

Décidabilité et complexité

- le problème de satisfaisabilité d'un concept est décidable :
  - PSPACE complet si la TBox est acyclique
  - EXPTIME complet en général

$\Rightarrow$ suite : procédure de décision via tableaux pour la satisfaisabilité d'une ABox

$\Rightarrow$ subsume tous les autres problèmes de raisonnement (pour des TBox acycliques)

La méthode des tableaux

- entrée : ABox $\mathcal{A}$
- sortie : ‘oui’ si $\mathcal{A}$ est satisfiable, ‘non’ sinon
- hypothèse : $\mathcal{A}$ en forme normale négative
  - négations seulement devant les atomes
- notions et notations :
  - configuration du tableau = ensemble fini d’ABox’es S
    - initialisation : $S = \{\mathcal{A}\}$
  - $S, \mathcal{A}$ à la place de $S \cup \mathcal{A}$
  - $\mathcal{A}[X]$ = X est un sous-ensemble de $\mathcal{A}$
  - format des règles :

$$\frac{S,\ \mathcal{A}[X]}{S,\ \mathcal{A}[Y_1], \ldots, \mathcal{A}[Y_n]}$$

    - si $X \subseteq \mathcal{A}$ alors ajouter Y_i à $\mathcal{A}$
    - applicable si $Y_1 \not\subseteq \mathcal{A}, \ldots$ et $Y_n \not\subseteq \mathcal{A}$

Règles de tableau

- règle pour $\sqcap$:

$$\frac{S,\ \mathcal{A}[(\mathtt{C} \sqcap \mathtt{D})(\mathtt{a})]}{S,\ \mathcal{A}[\mathtt{C}(\mathtt{a}),\mathtt{D}(\mathtt{a})]}$$

- règle pour $\sqcup$:

$$\frac{S,\ \mathcal{A}[(\mathtt{C} \sqcup \mathtt{D})(\mathtt{a})]}{S,\ \mathcal{A}[\mathtt{C}(\mathtt{a})],\ \mathcal{A}[\mathtt{D}(\mathtt{a})]}$$

- règle pour $\exists$:

$$\frac{S,\ \mathcal{A}[(\exists \mathtt{R}.\mathtt{C})(\mathtt{a})]}{S,\ \mathcal{A}[\mathtt{R}(\mathtt{a},\mathtt{b}),\mathtt{C}(\mathtt{b})]} \quad ((\mathtt{b}) \notin \mathcal{A})$$

b = nouveau individu

- règle pour $\forall$:

$$\frac{S,\ \mathcal{A}[(\forall \mathtt{R}.\mathtt{C})(\mathtt{a}),\mathtt{R}(\mathtt{a},\mathtt{b})]}{S,\ \mathcal{A}[\mathtt{C}(\mathtt{b})]}$$

Règles de tableau

- règle pour $\sqcap$:

$$\frac{S,\ \mathcal{A}[(\mathtt{C} \sqcap \mathtt{D})(\mathtt{a})]}{S,\ \mathcal{A}[\mathtt{C}(\mathtt{a}),\mathtt{D}(\mathtt{a})]}$$

- règle pour $\sqcup$:

$$\frac{S,\ \mathcal{A}[(\mathtt{C} \sqcup \mathtt{D})(\mathtt{a})]}{S,\ \mathcal{A}[\mathtt{C}(\mathtt{a})],\ \mathcal{A}[\mathtt{D}(\mathtt{a})]}$$

- règle pour $\exists$:

$$\frac{S,\ \mathcal{A}[(\exists \mathrm{R}.\mathrm{C})(\mathtt{a})]}{S,\ \mathcal{A}[\mathrm{R}(\mathtt{a},\mathtt{b}),\mathtt{C}(\mathtt{b})]} \quad ((\mathtt{b}) \notin \mathcal{A})$$

b = nouveau individu

- règle pour $\forall$:

$$\frac{S,\ \mathcal{A}[(\forall \mathrm{R}.\mathrm{C})(\mathtt{a}),\mathrm{R}(\mathtt{a},\mathtt{b})]}{S,\ \mathcal{A}[\mathtt{C}(\mathtt{b})]}$$

Tableaux : contradictions manifeste

Une ABox $\mathcal{A}$ contient une contradiction manifeste (*clash*) si il existe un a tel que

- $\bot(a) \in \mathcal{A}$, ou
- $C(a) \in \mathcal{A}$ et $\neg C(a) \in \mathcal{A}$

Sinon $\mathcal{A}$ est *ouvert*.

- exemple :
  $\mathcal{A} = \{\mathtt{Personne} \sqcap \exists\mathtt{parentDe.Personne} \sqcap \forall\mathtt{parentDe.}\bot(a)\}$
  1. déjà en forme normale négative
  2. $\{\{\mathtt{Personne}(a), \exists\mathtt{parentDe.Personne}(a), \forall\mathtt{parentDe.}\bot(a)\}\}$
     (règle pour $\sqcap$)
  3. $\{\{\mathtt{Personne}(a), \exists\mathtt{parentDe.Personne}(a), \forall\mathtt{parentDe.}\bot(a), \mathtt{Personne}(b)\}\}$
     (règle pour $\exists$)
  4. $\{\{\mathtt{Personne}(a), \exists\mathtt{parentDe.Personne}(a), \forall\mathtt{parentDe.}\bot(a), \mathtt{Personne}(b), \bot(b)\}\}$
     (règle pour $\forall$)
  5. contradiction manifeste : $\bot(b)$

Tableaux : contradictions manifeste

Une ABox $\mathcal{A}$ contient une contradiction manifeste (*clash*) si il existe un a tel que

- $\bot(\mathtt{a}) \in \mathcal{A}$, ou
- $\mathtt{C(a)} \in \mathcal{A}$ et $\neg\mathtt{C(a)} \in \mathcal{A}$

Sinon $\mathcal{A}$ est *ouvert*.

- exemple :
  $\mathcal{A} = \{\mathtt{Personne} \sqcap \exists\mathtt{parentDe.Personne} \sqcap \forall\mathtt{parentDe.}\bot\mathtt{(a)}\}$
  1. déjà en forme normale négative
  2. $\{\{\mathtt{Personne(a)}, \exists\mathtt{parentDe.Personne(a)}, \forall\mathtt{parentDe.}\bot\mathtt{(a)}\}\}$
     (règle pour $\sqcap$)
  3. $\{\{\mathtt{Personne(a)}, \exists\mathtt{parentDe.Personne(a)}, \forall\mathtt{parentDe.}\bot\mathtt{(a)},$
     $\mathtt{Personne(b)}\}\}$
     (règle pour $\exists$)
  4. $\{\{\mathtt{Personne(a)}, \exists\mathtt{parentDe.Personne(a)}, \forall\mathtt{parentDe.}\bot\mathtt{(a)},$
     $\mathtt{Personne(b)}, \bot\mathtt{(b)}\}\}$
     (règle pour $\forall$)
  5. contradiction manifeste : $\bot\mathtt{(b)}$

Raisonnement : exercice complet (1)

$$\boxed{\{\mathtt{A} \sqsubseteq \exists \mathtt{R.B}\} \cup \{\mathtt{A(a)}\} \models^? (\exists \mathtt{R.B})(\mathtt{a})}$$

(tâche = inférence de propriétés)

1. transformation en le problème de satisfaisabilité d'une ABox :

   $\boxed{\{\mathtt{A(a)}\} \cup \{(\neg\exists \mathtt{R.B})(\mathtt{a})\} \text{ satisfaisable p.r.à } \{\mathtt{A} \sqsubseteq \exists \mathtt{R.B}\} \text{ ?}}$

2. éliminer $\sqsubseteq$:

   $\boxed{\{\mathtt{A(a)}, (\neg\exists \mathtt{R.B})(\mathtt{a})\} \text{ satisfaisable p.r.à } \{\mathtt{A} \equiv \mathtt{A}' \sqcap \exists \mathtt{R.B}\} \text{ ?}}$

   $\Rightarrow$ TBox acyclique !

3. expansion de la ABox par la TBox : $\mathtt{A}^{\{\mathtt{A}\equiv\mathtt{A}'\sqcap\exists\mathtt{R.B}\}} = \mathtt{A}' \sqcap \exists \mathtt{R.B}$

   $\boxed{\{(\mathtt{A}' \sqcap \exists \mathtt{R.B})(\mathtt{a})\} \cup \{(\neg\exists \mathtt{R.B})(\mathtt{a})\} \text{ satisfaisable ?}}$

4. mise en forme normale négative :

   $\boxed{\{(\mathtt{A}' \sqcap \exists \mathtt{R.B})(\mathtt{a}), (\forall \mathtt{R}.\neg \mathtt{B})(\mathtt{a})\} \text{ satisfaisable ?}}$

5. construction d'un tableau . . .

Raisonnement : exercice complet (1)

$$\boxed{\{A \sqsubseteq \exists R.B\} \cup \{A(a)\} \models^{?} (\exists R.B)(a)}$$

(tâche = inférence de propriétés)

1. transformation en le problème de satisfaisabilité d'une ABox :

   $\{A(a)\} \cup \{(\neg\exists R.B)(a)\}$ satisfaisable p.r.à $\{A \sqsubseteq \exists R.B\}$?

2. éliminer $\sqsubseteq$:

   $\{A(a), (\neg\exists R.B)(a)\}$ satisfaisable p.r.à $\{A \equiv A' \sqcap \exists R.B\}$?

   $\Rightarrow$ TBox acyclique !

3. expansion de la ABox par la TBox : $A^{\{A \equiv A' \sqcap \exists R.B\}} = A' \sqcap \exists R.B$

   $\{(A' \sqcap \exists R.B)(a)\} \cup \{(\neg\exists R.B)(a)\}$ satisfaisable ?

4. mise en forme normale négative :

   $\{(A' \sqcap \exists R.B)(a), (\forall R.\neg B)(a)\}$ satisfaisable ?

5. construction d'un tableau . . .

Raisonnement : exercice complet (1)

$$\boxed{\{\mathtt{A} \sqsubseteq \exists \mathtt{R.B}\} \cup \{\mathtt{A(a)}\} \models^{?} (\exists \mathtt{R.B})(\mathtt{a})}$$

(tâche = inférence de propriétés)

1. transformation en le problème de satisfaisabilité d'une ABox :

   $\{\mathtt{A(a)}\} \cup \{(\neg\exists \mathtt{R.B})(\mathtt{a})\}$ satisfaisable p.r.à $\{\mathtt{A} \sqsubseteq \exists \mathtt{R.B}\}$?

2. éliminer $\sqsubseteq$:

   $\{\mathtt{A(a)}, (\neg\exists \mathtt{R.B})(\mathtt{a})\}$ satisfaisable p.r.à $\{\mathtt{A} \equiv \mathtt{A'} \sqcap \exists \mathtt{R.B}\}$?

   $\Rightarrow$ TBox acyclique !

3. expansion de la ABox par la TBox : $\mathtt{A}^{\{\mathtt{A}\equiv\mathtt{A'}\sqcap\exists\mathtt{R.B}\}} = \mathtt{A'} \sqcap \exists \mathtt{R.B}$

   $\{(\mathtt{A'} \sqcap \exists \mathtt{R.B})(\mathtt{a})\} \cup \{(\neg\exists \mathtt{R.B})(\mathtt{a})\}$ satisfaisable ?

4. mise en forme normale négative :

   $\{(\mathtt{A'} \sqcap \exists \mathtt{R.B})(\mathtt{a}), (\forall \mathtt{R}.\neg \mathtt{B})(\mathtt{a})\}$ satisfaisable ?

5. construction d'un tableau ...

Raisonnement : exercice complet (1)

$$\boxed{\{\mathtt{A} \sqsubseteq \exists \mathtt{R.B}\} \cup \{\mathtt{A(a)}\} \models^{?} (\exists \mathtt{R.B})(\mathtt{a})}$$

(tâche = inférence de propriétés)

1. transformation en le problème de satisfaisabilité d'une ABox :

   $\boxed{\{\mathtt{A(a)}\} \cup \{(\neg\exists \mathtt{R.B})(\mathtt{a})\} \text{ satisfaisable p.r.à } \{\mathtt{A} \sqsubseteq \exists \mathtt{R.B}\} \text{ ?}}$

2. éliminer $\sqsubseteq$:

   $\boxed{\{\mathtt{A(a)}, (\neg\exists \mathtt{R.B})(\mathtt{a})\} \text{ satisfaisable p.r.à } \{\mathtt{A} \equiv \mathtt{A}' \sqcap \exists \mathtt{R.B}\} \text{ ?}}$

   $\Rightarrow$ TBox acyclique !

3. expansion de la ABox par la TBox : $\mathtt{A}^{\{\mathtt{A}\equiv\mathtt{A}'\sqcap\exists\mathtt{R.B}\}} = \mathtt{A}' \sqcap \exists \mathtt{R.B}$

   $\boxed{\{(\mathtt{A}' \sqcap \exists \mathtt{R.B})(\mathtt{a})\} \cup \{(\neg\exists \mathtt{R.B})(\mathtt{a})\} \text{ satisfaisable ?}}$

4. mise en forme normale négative :

   $\boxed{\{(\mathtt{A}' \sqcap \exists \mathtt{R.B})(\mathtt{a}), (\forall \mathtt{R}.\neg \mathtt{B})(\mathtt{a})\} \text{ satisfaisable ?}}$

5. construction d'un tableau ...

Raisonnement : exercice complet (1)

$$\boxed{\{A \sqsubseteq \exists R.B\} \cup \{A(a)\} \models^{?} (\exists R.B)(a)}$$

(tâche = inférence de propriétés)

1. transformation en le problème de satisfaisabilité d'une ABox :

   $\boxed{\{A(a)\} \cup \{(\neg\exists R.B)(a)\} \text{ satisfaisable p.r.à } \{A \sqsubseteq \exists R.B\} \text{ ?}}$

2. éliminer $\sqsubseteq$:

   $\boxed{\{A(a), (\neg\exists R.B)(a)\} \text{ satisfaisable p.r.à } \{A \equiv A' \sqcap \exists R.B\} \text{ ?}}$

   $\Rightarrow$ TBox acyclique !

3. expansion de la ABox par la TBox : $A^{\{A \equiv A' \sqcap \exists R.B\}} = A' \sqcap \exists R.B$

   $\boxed{\{(A' \sqcap \exists R.B)(a)\} \cup \{(\neg\exists R.B)(a)\} \text{ satisfaisable ?}}$

4. mise en forme normale négative :

   $\boxed{\{(A' \sqcap \exists R.B)(a), (\forall R.\neg B)(a)\} \text{ satisfaisable ?}}$

5. construction d'un tableau . . .

Raisonnement : exercice complet (2)

$\boxed{\{(A' \sqcap \exists R.B)(a), (\forall R.\neg B)(a)\} \text{ satisfaisable ?}}$

construction du tableau :

1. $\{\{(A' \sqcap \exists R.B)(a), (\forall R.\neg B)(a)\}\}$
2. $\{\{A'(a), (\exists R.B)(a), (\forall R.\neg B)(a)\}\}$ (règle pour $\sqcap$)
3. $\{\{A'(a), (\exists R.B)(a), (\forall R.\neg B)(a), R(a,b), B(b)\}\}$
   (règle pour $\exists$; b nouveau)
4. $\{\{A'(a), (\exists R.B)(a), (\forall R.\neg B)(a), R(a,b), B(b), (\neg B)(b)\}\}$
   (règle pour $\forall$)

$\Rightarrow$ contradiction manifeste

$\Rightarrow$ la ABox $\{(A' \sqcap \exists R.B)(a), (\forall R.\neg B)(a)\}$ insatisfaisable

$\Rightarrow$ la TBox $\{A \equiv A' \sqcap \exists R.B\} \cup \{A(a)\}$ a comme conséquence que a a la propriété $\exists R.B$:

- $\{A \equiv A' \sqcap \exists R.B\} \cup \{A(a)\} \models (\exists R.B)(a)$

Raisonnement : exercice complet (2)

$\boxed{\{(A' \sqcap \exists R.B)(a), (\forall R.\neg B)(a)\} \text{ satisfaisable ?}}$

construction du tableau :

1. $\{\{(A' \sqcap \exists R.B)(a), (\forall R.\neg B)(a)\}\}$
2. $\{\{A'(a), (\exists R.B)(a), (\forall R.\neg B)(a)\}\}$ (règle pour $\sqcap$)
3. $\{\{A'(a), (\exists R.B)(a), (\forall R.\neg B)(a), R(a,b), B(b)\}\}$ (règle pour $\exists$; b nouveau)
4. $\{\{A'(a), (\exists R.B)(a), (\forall R.\neg B)(a), R(a,b), B(b), (\neg B)(b)\}\}$ (règle pour $\forall$)

$\Rightarrow$ contradiction manifeste

$\Rightarrow$ la ABox $\{(A' \sqcap \exists R.B)(a), (\forall R.\neg B)(a)\}$ insatisfaisable

$\Rightarrow$ la TBox $\{A \equiv A' \sqcap \exists R.B\} \cup \{A(a)\}$ a comme conséquence que a a la propriété $\exists R.B$:

- $\{A \equiv A' \sqcap \exists R.B\} \cup \{A(a)\} \models (\exists R.B)(a)$

Raisonnement : exercice complet (2)

$\boxed{\{(A' \sqcap \exists R.B)(a), (\forall R.\neg B)(a)\} \text{ satisfaisable ?}}$

construction du tableau :

1. $\{\{(A' \sqcap \exists R.B)(a), (\forall R.\neg B)(a)\}\}$
2. $\{\{A'(a), (\exists R.B)(a), (\forall R.\neg B)(a)\}\}$ (règle pour $\sqcap$)
3. $\{\{A'(a), (\exists R.B)(a), (\forall R.\neg B)(a), R(a,b), B(b)\}\}$ (règle pour $\exists$; b nouveau)
4. $\{\{A'(a), (\exists R.B)(a), (\forall R.\neg B)(a), R(a,b), B(b), (\neg B)(b)\}\}$ (règle pour $\forall$)

$\Rightarrow$ contradiction manifeste

$\Rightarrow$ la ABox $\{(A' \sqcap \exists R.B)(a), (\forall R.\neg B)(a)\}$ insatisfaisable

$\Rightarrow$ la TBox $\{A \equiv A' \sqcap \exists R.B\} \cup \{A(a)\}$ a comme conséquence que a a la propriété $\exists R.B$:

- $\{A \equiv A' \sqcap \exists R.B\} \cup \{A(a)\} \models (\exists R.B)(a)$

Raisonnement : exercice complet (2)

$\boxed{\{(A' \sqcap \exists R.B)(a), (\forall R.\neg B)(a)\} \text{ satisfaisable ?}}$

construction du tableau :

1. $\{\{(A' \sqcap \exists R.B)(a), (\forall R.\neg B)(a)\}\}$
2. $\{\{A'(a), (\exists R.B)(a), (\forall R.\neg B)(a)\}\}$ (règle pour $\sqcap$)
3. $\{\{A'(a), (\exists R.B)(a), (\forall R.\neg B)(a), R(a,b), B(b)\}\}$ (règle pour $\exists$; b nouveau)
4. $\{\{A'(a), (\exists R.B)(a), (\forall R.\neg B)(a), R(a,b), B(b), (\neg B)(b)\}\}$ (règle pour $\forall$)

$\Rightarrow$ contradiction manifeste

$\Rightarrow$ la ABox $\{(A' \sqcap \exists R.B)(a), (\forall R.\neg B)(a)\}$ insatisfaisable

$\Rightarrow$ la TBox $\{A \equiv A' \sqcap \exists R.B\} \cup \{A(a)\}$ a comme conséquence que a a la propriété $\exists R.B$:

- $\{A \equiv A' \sqcap \exists R.B\} \cup \{A(a)\} \models (\exists R.B)(a)$

Raisonnement : exercice complet (2)

$\{(A' \sqcap \exists R.B)(a), (\forall R.\neg B)(a)\}$ satisfaisable ?

construction du tableau :

1. $\{\{(A' \sqcap \exists R.B)(a), (\forall R.\neg B)(a)\}\}$
2. $\{\{A'(a), (\exists R.B)(a), (\forall R.\neg B)(a)\}\}$ (règle pour $\sqcap$)
3. $\{\{A'(a), (\exists R.B)(a), (\forall R.\neg B)(a), R(a,b), B(b)\}\}$
   (règle pour $\exists$; b nouveau)
4. $\{\{A'(a), (\exists R.B)(a), (\forall R.\neg B)(a), R(a,b), B(b), (\neg B)(b)\}\}$
   (règle pour $\forall$)

$\Rightarrow$ contradiction manifeste

$\Rightarrow$ la ABox $\{(A' \sqcap \exists R.B)(a), (\forall R.\neg B)(a)\}$ insatisfaisable

$\Rightarrow$ la TBox $\{A \equiv A' \sqcap \exists R.B\} \cup \{A(a)\}$ a comme conséquence que a a la propriété $\exists R.B$:

- $\{A \equiv A' \sqcap \exists R.B\} \cup \{A(a)\} \models (\exists R.B)(a)$

Raisonnement : exercice complet (2)

$\boxed{\{(\mathtt{A}' \sqcap \exists \mathtt{R}.\mathtt{B})(\mathtt{a}), (\forall \mathtt{R}.\neg \mathtt{B})(\mathtt{a})\} \text{ satisfaisable ?}}$

construction du tableau :

1. $\{\{(\mathtt{A}' \sqcap \exists \mathtt{R}.\mathtt{B})(\mathtt{a}), (\forall \mathtt{R}.\neg \mathtt{B})(\mathtt{a})\}\}$
2. $\{\{\mathtt{A}'(\mathtt{a}), (\exists \mathtt{R}.\mathtt{B})(\mathtt{a}), (\forall \mathtt{R}.\neg \mathtt{B})(\mathtt{a})\}\}$ (règle pour $\sqcap$)
3. $\{\{\mathtt{A}'(\mathtt{a}), (\exists \mathtt{R}.\mathtt{B})(\mathtt{a}), (\forall \mathtt{R}.\neg \mathtt{B})(\mathtt{a}), \mathtt{R}(\mathtt{a},\mathtt{b}), \mathtt{B}(\mathtt{b})\}\}$
   (règle pour $\exists$; b nouveau)
4. $\{\{\mathtt{A}'(\mathtt{a}), (\exists \mathtt{R}.\mathtt{B})(\mathtt{a}), (\forall \mathtt{R}.\neg \mathtt{B})(\mathtt{a}), \mathtt{R}(\mathtt{a},\mathtt{b}), \mathtt{B}(\mathtt{b}), (\neg \mathtt{B})(\mathtt{b})\}\}$
   (règle pour $\forall$)

$\Rightarrow$ contradiction manifeste

$\Rightarrow$ la ABox $\{(\mathtt{A}' \sqcap \exists \mathtt{R}.\mathtt{B})(\mathtt{a}), (\forall \mathtt{R}.\neg \mathtt{B})(\mathtt{a})\}$ insatisfaisable

$\Rightarrow$ la TBox $\{\mathtt{A} \equiv \mathtt{A}' \sqcap \exists \mathtt{R}.\mathtt{B}\} \cup \{\mathtt{A}(\mathtt{a})\}$ a comme conséquence que a a la propriété $\exists \mathtt{R}.\mathtt{B}$:

- $\{\mathtt{A} \equiv \mathtt{A}' \sqcap \exists \mathtt{R}.\mathtt{B}\} \cup \{\mathtt{A}(\mathtt{a})\} \models (\exists \mathtt{R}.\mathtt{B})(\mathtt{a})$

Raisonnement : exercice complet (2)

$\boxed{\{(\mathtt{A}' \sqcap \exists \mathtt{R}.\mathtt{B})(\mathtt{a}), (\forall \mathtt{R}.\neg \mathtt{B})(\mathtt{a})\} \text{ satisfaisable ?}}$

construction du tableau :

1. $\{\{(\mathtt{A}' \sqcap \exists \mathtt{R}.\mathtt{B})(\mathtt{a}), (\forall \mathtt{R}.\neg \mathtt{B})(\mathtt{a})\}\}$
2. $\{\{\mathtt{A}'(\mathtt{a}), (\exists \mathtt{R}.\mathtt{B})(\mathtt{a}), (\forall \mathtt{R}.\neg \mathtt{B})(\mathtt{a})\}\}$ (règle pour $\sqcap$)
3. $\{\{\mathtt{A}'(\mathtt{a}), (\exists \mathtt{R}.\mathtt{B})(\mathtt{a}), (\forall \mathtt{R}.\neg \mathtt{B})(\mathtt{a}), \mathtt{R}(\mathtt{a},\mathtt{b}), \mathtt{B}(\mathtt{b})\}\}$
   (règle pour $\exists$; b nouveau)
4. $\{\{\mathtt{A}'(\mathtt{a}), (\exists \mathtt{R}.\mathtt{B})(\mathtt{a}), (\forall \mathtt{R}.\neg \mathtt{B})(\mathtt{a}), \mathtt{R}(\mathtt{a},\mathtt{b}), \mathtt{B}(\mathtt{b}), (\neg \mathtt{B})(\mathtt{b})\}\}$
   (règle pour $\forall$)

$\Rightarrow$ contradiction manifeste

$\Rightarrow$ la ABox $\{(\mathtt{A}' \sqcap \exists \mathtt{R}.\mathtt{B})(\mathtt{a}), (\forall \mathtt{R}.\neg \mathtt{B})(\mathtt{a})\}$ insatisfaisable

$\Rightarrow$ la TBox $\{\mathtt{A} \equiv \mathtt{A}' \sqcap \exists \mathtt{R}.\mathtt{B}\} \cup \{\mathtt{A}(\mathtt{a})\}$ a comme conséquence que a a la propriété $\exists \mathtt{R}.\mathtt{B}$:

- $\{\mathtt{A} \equiv \mathtt{A}' \sqcap \exists \mathtt{R}.\mathtt{B}\} \cup \{\mathtt{A}(\mathtt{a})\} \models (\exists \mathtt{R}.\mathtt{B})(\mathtt{a})$

Raisonnement : un autre exercice complet (1)

$$\boxed{\{\mathtt{A} \equiv (\exists \mathrm{R}.\mathrm{B}_1) \sqcap (\exists \mathrm{R}.\mathrm{B}_2)\} \models^? \mathtt{A} \sqsubseteq \exists \mathrm{R}.(\mathrm{B}_1 \sqcap \mathrm{B}_2)}$$

(tâche = subsomption de concepts p.r.à une TBox)

1. transformation en problème de satisfaisabilité de concept p.r.à une TBox

$$\boxed{\mathtt{A} \sqcap \neg\exists \mathrm{R}.(\mathrm{B}_1 \sqcap \mathrm{B}_2) \text{ satisfaisable p.r.à } \{\mathtt{A} \equiv (\exists \mathrm{R}.\mathrm{B}_1) \sqcap (\exists \mathrm{R}.\mathrm{B}_2)\} \text{ ?}}$$

2. expansion de la ABox par la TBox :

$$\boxed{(\exists \mathrm{R}.\mathrm{B}_1) \sqcap (\exists \mathrm{R}.\mathrm{B}_2) \sqcap \neg\exists \mathrm{R}.(\mathrm{B}_1 \sqcap \mathrm{B}_2) \text{ satisfaisable ?}}$$

3. mise en forme normale négative :

$$\boxed{(\exists \mathrm{R}.\mathrm{B}_1) \sqcap (\exists \mathrm{R}.\mathrm{B}_2) \sqcap \forall \mathrm{R}.(\neg \mathrm{B}_1 \sqcup \neg \mathrm{B}_2) \text{ satisfaisable ?}}$$

4. construction d'un tableau . . .

Raisonnement : un autre exercice complet (2)

$\boxed{(\exists R.B_1) \sqcap (\exists R.B_2) \sqcap \forall R.(\neg B_1 \sqcup \neg B_2) \text{ satisfaisable ?}}$

construction du tableau :

1. $\{ \{(\exists R.B_1) \sqcap (\exists R.B_2) \sqcap \forall R.(\neg B_1 \sqcup \neg B_2)(a)\} \}$
2. $\{ \{(\exists R.B_1)(a), (\exists R.B_2)(a), (\forall R.(\neg B_1 \sqcup \neg B_2))(a)\} \}$

   (règle pour $\sqcap$, deux fois)
3. $\{ \{(\exists R.B_1)(a), (\exists R.B_2)(a), (\forall R.(\neg B_1 \sqcup \neg B_2))(a),$
   $R(a,b_1), B_1(b_1), R(a,b_2), B_2(b_2)\} \}$ (règle pour $\exists$, deux fois)
4. $\{ \{(\exists R.B_1)(a), (\exists R.B_2)(a), (\forall R.(\neg B_1 \sqcup \neg B_2))(a),$
   $R(a,b_1), B_1(b_1), R(a,b_2), B_2(b_2),$
   $(\neg B_1 \sqcup \neg B_2)(b_1), (\neg B_1 \sqcup \neg B_2)(b_2)\} \} = \{\mathcal{A}_0\}$

   (règle pour $\forall$, deux fois)
5. $\{ \mathcal{A}_0 \cup \{(\neg B_1)(b_1)\}, \mathcal{A}_0 \cup \{(\neg B_2)(b_1)\} \}$ (règle pour $\sqcup$)
6. $\{ \mathcal{A}_0 \cup \{(\neg B_1)(b_1)\} \cup \{(\neg B_1)(b_2)\}, \mathcal{A}_0 \cup \{(\neg B_1)(b_1)\} \cup \{(\neg B_2)(b_2)\},$
   $\mathcal{A}_0 \cup \{(\neg B_2)(b_1)\} \cup \{(\neg B_1)(b_2)\}, \mathcal{A}_0 \cup \{(\neg B_2)(b_1)\} \cup \{(\neg B_2)(b_2)\} \}$

   (règle pour $\sqcup$, deux fois)

$\Rightarrow$ la troisième ABox est ouverte $\Rightarrow$ satisfaisable

Tableaux pour ALC : la grande finale

- La configuration S est satisfaisable ssi il existe une ABox $\mathcal{A}_i \in S$ et une interprétation $\mathcal{I}$ t.q. $\mathcal{I} \Vdash \mathcal{A}_i$.
- La configuration S est saturée ssi plus aucune règle ne peut être appliquée

Proposition (terminaison)

Pour toute entrée $\mathcal{A}$, la procédure de construction de tableau termine par une configuration saturée.

Proposition (adéquation)

Si la ABox $\mathcal{A}$ est satisfaisable alors toutes les configurations obtenues à partir de $\{\mathcal{A}\}$ sont ouvertes.

Proposition (complétude)

Si S est une configuration saturée et ouverte alors il existe une ABox $\mathcal{A} \in S$ tel que $\mathcal{A}$ est satisfaisable.

Tableaux pour des extensions de ALC

- ALCN
- inclusion de concepts généraux
  - requiert test de boucle
  - EXPTIME complet
- inclusion de rôles
- rôles transitifs
- réstrictions de cardinalité qualifiées
  - indécidable si combiné avec intersection de rôles !
  - indécidable si combiné avec rôles transitifs !
- constructeur 'un-parmi $a_1, \ldots, a_n$' : $\{a_1, \ldots, a_n\}$
  - $(\{a_1, \ldots, a_n\})^{\mathcal{I}} = \{a_1^{\mathcal{I}}, \ldots, a_n^{\mathcal{I}}\}$
  - exemple : $\mathtt{Personne} \equiv \{\mathtt{Alice}, \mathtt{Bob}, \mathtt{Charles}\}$
- domaines concrètes
  - nombres entiers : $\mathtt{Adulte} \equiv \mathtt{Personne} \sqcap \exists \mathtt{ag\acute{e}eDe}.{\geq}18$
  - ...

Conclusion sur la partie DL

- DL = représenter + raisonner
- DL = plusieurs logiques
  - expressivité (quelles réstrictions du langage permis par le domaine ?)
  - complexité (P, NP, PSPACE, EXPTIME)
- DL base de OWL (OWL-DL)