

Vers une approche de raisonnement hybride entre matérialisation et virtualisation

Encadrants : Federico Ulliana, Michel Leclerc, Pierre Bisquert

David CAMARAZO

16 juin 2022

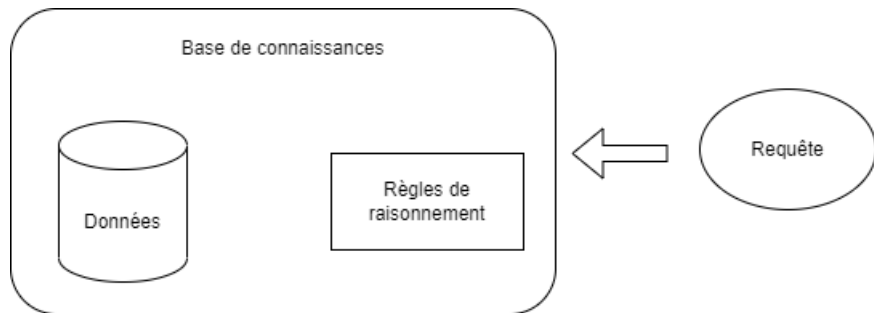
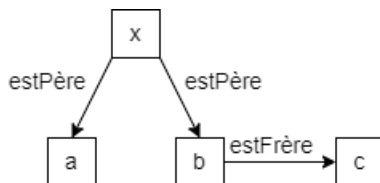


Figure – base de connaissances



$\exists x(\text{estPère}(x,a) \wedge \text{estPère}(x,b) \wedge \text{estFrère}(b,c))$

Figure – Base de faits

Base de règles

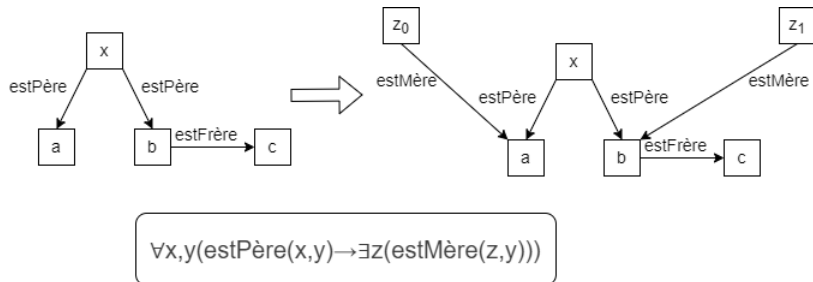


Figure – Base de règles



Figure – Illustration de la marche avant

Raisonner en marche avant

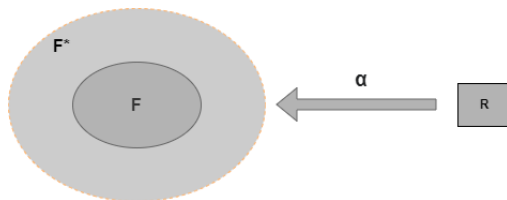


Figure – Illustration de la marche avant

Raisonner en marche avant

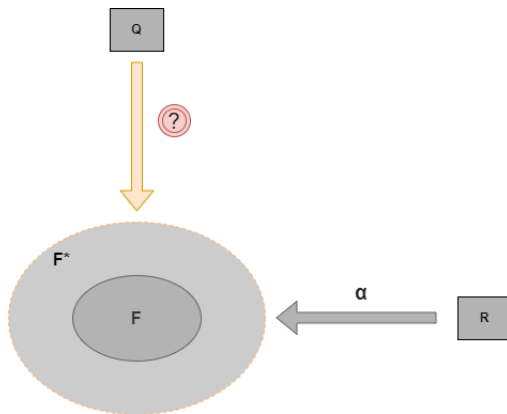


Figure – Illustration de la marche avant

Raisonner en marche arrière

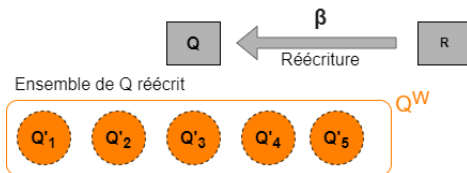


Figure – Illustration de la marche arrière

Raisonner en marche arrière

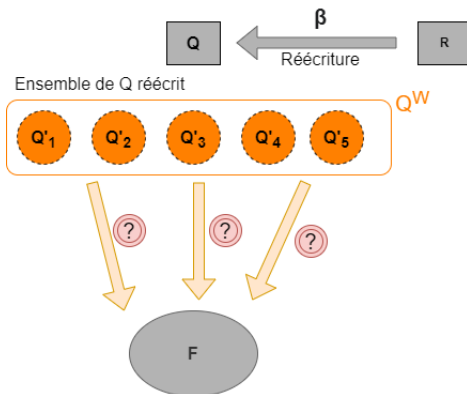


Figure – Illustration de la marche arrière

- Marche avant produit trop de faits.
- Marche arrière n'est pas toujours possible.

Solution ?

On propose une solution hybride.

Objectif de notre stratégie

Objectif

On suppose que l'on connaît les informations interrogées par les utilisateurs.
On souhaite alors matérialiser ces informations afin qu'elles puissent être interrogées rapidement. Le reste est virtualisé.

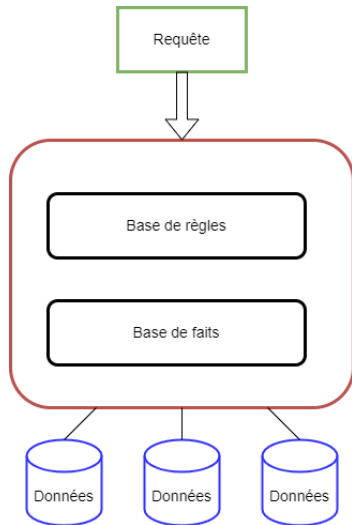


Figure – Illustration d'un besoin stratégie hybride

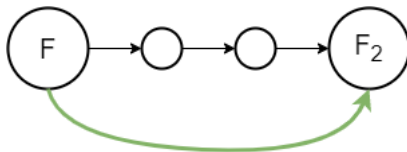


Figure – Raccourci dans la dérivation

β -composition de règles existentielles

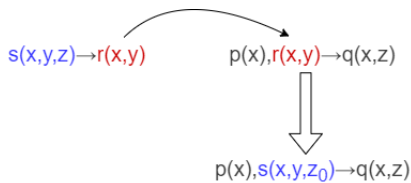


Figure – Illustration de la composition de règles

\mathcal{C} -composition

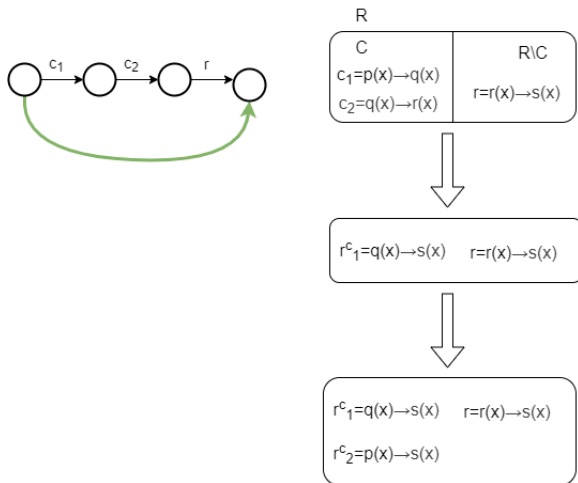


Figure – Illustration de la \mathcal{C} -composition

Problème lié à l'unification

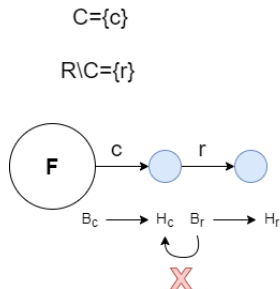


Figure – Problème apportée par les règles existentielles

Exemple de problème avec les existentielles

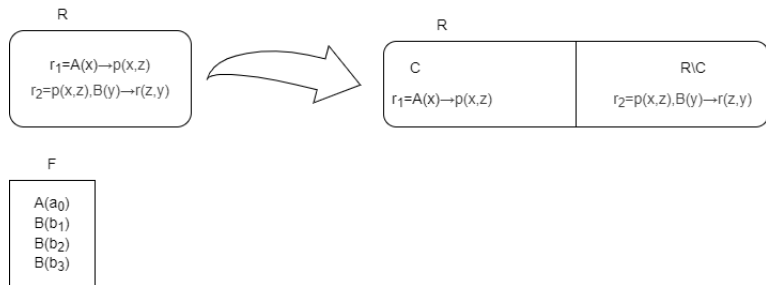


Figure – Exemple de problème apportée par les règles existentielles

Exemple de problème avec les existentielles

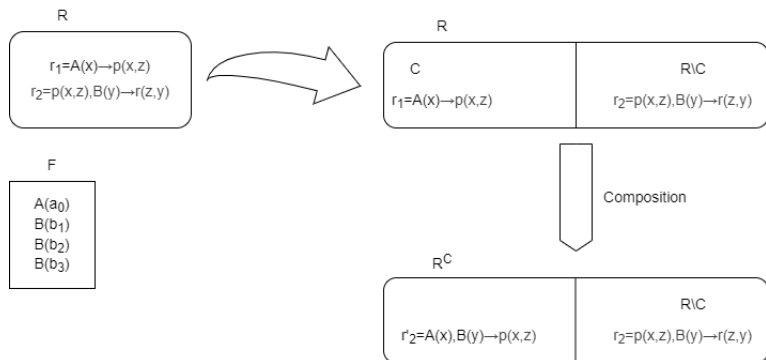


Figure – Exemple de problème apportée par les règles existentielles

Exemple de problème avec les existentielles

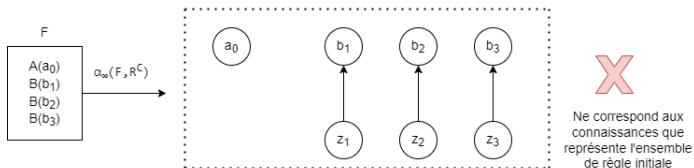
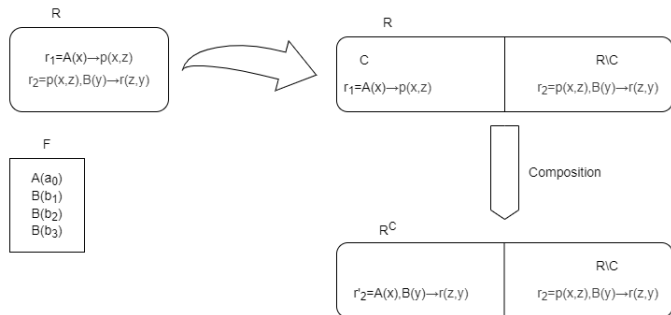


Figure – Exemple de problème apportée par les règles existentielles

Exemple de problème avec les existentielles

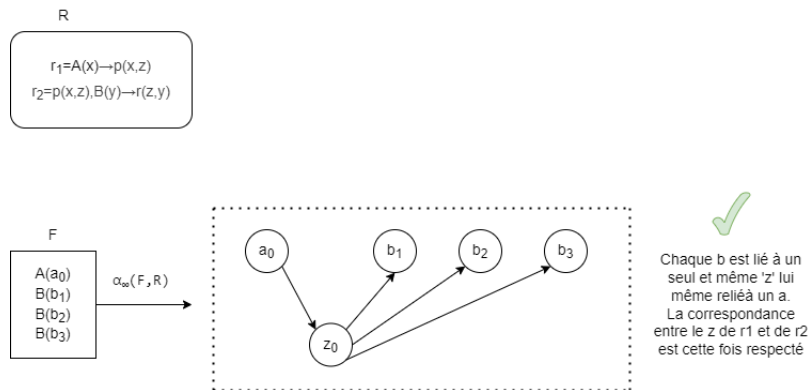


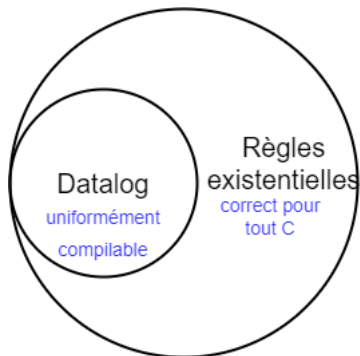
Figure – Exemple de problème apportée par les règles existentielles

Compilable

On dit qu'un couple $(\mathcal{R}, \mathcal{C})$ est compilable si et seulement si :

Pour toutes bases de faits \mathcal{F} et pour toute requête Q , il existe deux entiers k_1 et k_2 tels que :

$$\mathcal{F}, \mathcal{R} \models Q \iff \alpha_{k_1}(\mathcal{F}, \mathcal{R}^c) \models \beta_{k_2}(Q, \mathcal{C})$$



Comparaison avec la marche avant

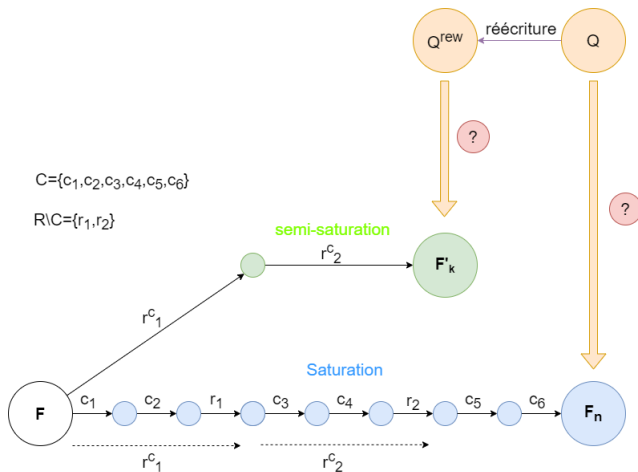


Figure – Illustration de la semi-saturation

Conclusion

- La marche avant demande beaucoup de mémoire pour enregistrer des faits inutiles
- La marche arrière demande beaucoup de calculs pour retrouver l'information

Solution ?

Nous proposons une stratégie de raisonnement appelée la semi-saturation qui ne matérialise que les faits les plus important.

La dérivation semi-saturée a pour but de réduire le nombre de faits à enregistrer. Elle permet également, si la réécriture est nécessaire, de réduire l'ensemble de règles utilisé pour la réécriture.

- Travailler sur la caractérisation des ensembles compilables
- Définition d'une fonction de coût qui estime le coût de marche avant par rapport à marche arrière. Il faudra alors se poser la question des critères sur lesquels se basent cette fonction de coût. Le temps nécessaire au calcul ? La mémoire nécessaire ?