logique et
REPRÉSENTATION
des connaissances

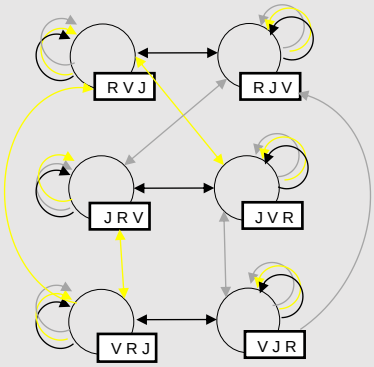
Compte-rendu du septième TME : lecture et interprétation de graphes épistémiques sous Haskell.

28 novembre 2019

L////// S///////, R////// M///////

Interprétation du modèle hexa

Le modèle ci-contre représente une distribution de cartes. Il est chargé en mémoire sous le nom `model0`. Soit (r,j,v) le monde où A a la carte rouge, B la carte jaune, C la carte verte ; les autres mondes sont construits selon le même système.
Les agent.e.s n'ont pas d'accès immédiat aux cartes des autres, mais se prêtent au jeu des devinettes, guidé.e.s par les annonces publiques de chacun.e.



↑ Modèle initial décrit en Haskell par
Mo [('r','v','j'), ('r','j','v'), ('j','r','v'), ('j','v','r'), ('v','r','j'), ('v','j','r')] [a,b,c] []
[(a, [[('r','v','j'),('r','j','v')],
[('j','r','v'),('j','v','r')],
[('v','r','j'),('v','j','r')]]),
(b, [[('r','v','j'),('j','v','r')],
[('r','j','v'),('v','j','r')],
[('v','r','j'),('j','r','v')]]),
(c, [[('r','v','j'),('v','r','j')],
[('j','v','r'),('v','j','r')],
[('j','r','v'),('r','j','v')]])]
[]

La première partie de la définition énumère les mondes du modèle. La deuxième liste les paires de mondes qui sont réciproquement accessibles selon les relations disponibles en passant la réflexivité et la symétrie sous silence.

La première annonce publique traduit une situation impossible. A dit savoir que B a la carte verte. A supposer que chacun.e part d'une représentation juste de la réalité et supprime les mondes non concernés, les agent.e.s se retrouvent tou.te.s avec des relations vides : A n'avait accès qu'à des paires de mondes où iel hésite sur la carte de B, et à entendre son annonce, on est certain.e qu'iel ment.

`upd_pa model0 (Kn a holds_b_verte)`
Mo [] [a,b,c] [] [(a,[]),(b,[]),(c,[])] []

La seconde annonce publique est une annonce non informative. A exprime qu'iel ne sait pas que B porte la carte verte, ou encore que chaque monde accessible à A a au moins un successeur où B n'a pas la carte verte. Or, c'est déjà vrai au départ. Personne ne peut mettre ses croyances à jour.

`upd_pa model0 (Ng(Kn a holds_b_verte))`
(pas de différence)

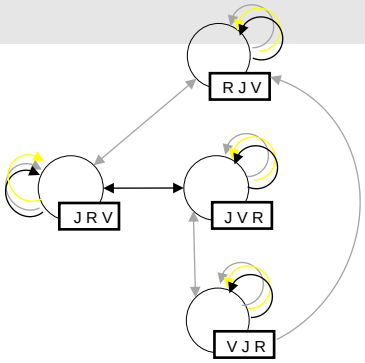
Reprenons du départ.
La première série d'annonces séquentielles permet bien de restreindre le modèle à un seul monde, auquel tous ont accès par réflexivité. Plus personne n'hésite sur la situation réelle.

```
model1 = upd_pa model0 (Kn b (Ng(holds_b_jaune)))
  model2 = upd_pa model1 (Ng(Kn a holds_b_rouge))
model3 = upd_pa model2 (Kn a (Ng(holds_a_jaune)))
```
Mo [('r','v','j')] [a,b,c] [] [(a,[[('r','v','j')]]),(b,[[('r','v','j')]]),(c,[[('r','v','j')]])] []

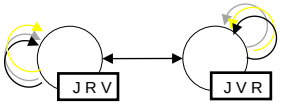
A a la carte rouge, B la carte verte, et C écope de la carte jaune, et chacun.e le sait.

On réinitialise.
La deuxième série d'annonces séquentielles a le même effet (bien que le monde final soit bien sûr différent). Pas à pas, détail ci-contre,

C dit qu'iel n'a pas la carte jaune,
`model1 = upd_pa model0 (Kn c (Ng(holds_c_jaune)))`
et tous les mondes concernés disparaissent:
Mo [('r','j','v'),('j','r','v'),('j','v','r'),('v','j','r')] [a,b,c] []
[(a,[[('r','j','v')],[('j','r','v'),('j','v','r')],[('v','j','r')]]),
(b,[[('j','v','r')],[('r','j','v'),('v','j','r')],[('j','r','v')]]),
(c,[[('j','v','r'),('v','j','r')],[('j','r','v'),('r','j','v')]])] []



A, qu'iel ne sait pas que B serait conscient.e d'avoir la carte jaune (s'iel l'avait !)
`model2 = upd_pa model1 (Ng(Kn a (Kn b(holds_b_jaune))))`,
et la paire de mondes concernée disparaît:
Mo [('j','r','v'),('j','v','r')] [a,b,c] []
[(a,[[('j','r','v'),('j','v','r')]]),
(b,[[('j','v','r')],[('j','r','v')]]),
(c,[[('j','v','r')],[('j','r','v')]])] []



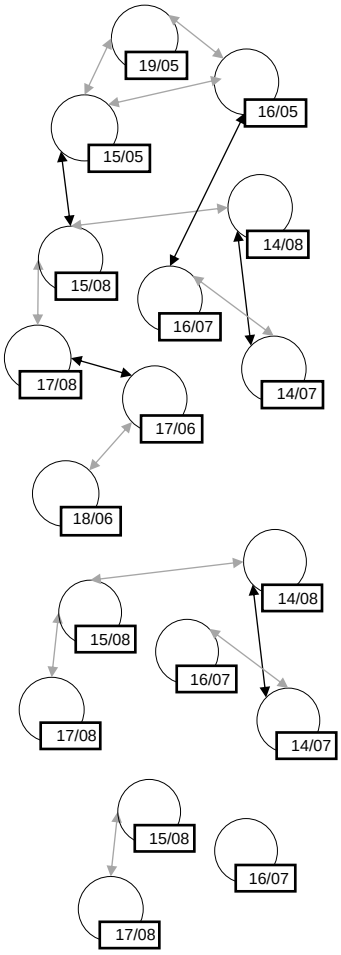
et C, de nouveau, que B n'a pas la carte verte,
`model3 = upd_pa model2 (Kn c (Ng(holds_b_verte)))`
et tous les mondes concernés disparaissent pour laisser
Mo [('j','r','v')] [a,b,c] [] [(a,[[('j','r','v')]]),(b,[[('j','r','v')]]),(c,[[('j','r','v')]])] []

A a la carte jaune, B la carte rouge, et C écope de la carte verte, et chacun.e le sait.

De telles déductions sont soumises à l'ordre de publication des informations - on le constate en mélangeant les annonces de la première séquence. Supprimer un monde très « ouvert » (qui donne accès à beaucoup d'autres) en premier permet de réduire très vite les possibilités et donc de se retrouver facilement comme on l'espère avec un modèle réduit à un singleton. Si ce monde est supprimé plus tard, la convergence ne peut pas être atteinte avec le même nombre d'étapes. Il faut plus d'informations pour rayer successivement des mondes que l'on avait pu éliminer d'un coup.

L'anniversaire de Cheryl

Cheryl décide de poser une colle à ses amis. Sur dix dates possibles, Albert (en gris) connaît le mois réel, et Bernard (en noir) le jour réel de son anniversaire. En notant (d/m) le monde où l'anniversaire tombe le d[ème] jour du mois m, le modèle s'exprime par le graphe ci-contre où, on nous pardonnera, on ne représente pas les arcs de réflexivité.



Nous définissons le fait de connaître la date d'anniversaire de Cheryl comme l'obtention d'un modèle de Kripke contenant un seul monde, un couple formé d'un jour et d'un mois. C'est conforme à la définition lue dans le fichier `tme4-cheryl.hs` :

```
knWhich :: Agent -> Form (Int, [Char])
knWhich i = Disj [ Kn i (Info s) | s <- allDays ]
```

L'enquête d'Albert et Bernard va aussi être orientée par une série d'annonces publiques. Voyons si elles permettent d'isoler un seul monde pour un observateur extérieur.

Albert déclare [ne pas connaître la date d'anniversaire], et sait que Bernard non plus.
`model1 = upd_pa model0 Kn a (Ng(knWhich b)))`*
Mo [(14,"July"),(16,"July"),(14,"August"),(15,"August"),(17,"August")] [a,b] []
[(a, [[(14,"July"),(16,"July")], [(14,"August"),(15,"August"),(17,"August")]]),
(b, [[(14,"July"),(14,"August")], [(15,"August")], [(16,"July")], [(17,"August")]])] []
Disparaissent alors les mondes où le jour suffirait à avoir le mois).
Ce ne peut donc être ni en mai, ni en juin…

…et d'un coup Bernard n'hésite plus :
`model2 = upd_pa model1 (knWhich b)`
Mo [(16,"July"),(15,"August"),(17,"August")] [a,b] []
[(a,[[(16,"July")],[(15,"August"),(17,"August")]]),
(b,[[(15,"August")],[(16,"July")],[(17,"August")]])] []
Bien que trois mondes soient encore disponibles, aucun n'a de successeur pour lui. S'il est dans cette situation, c'est que le jour qu'on lui a confié n'est disponible que sur un seul des deux mois. Ce n'est donc pas le 14. Bernard a terminé l'enquête : comme il sait le jour, il n'a plus qu'un monde possible dans son esprit, le bon - mais Albert qui ne le sait pas va pouvoir accéder au 15 août, au 17 août, et au 16 juillet, et nous aussi…

Soudain, Albert n'hésite plus non plus :
`model3 = upd_pa model2 (knWhich a)`
Mo [(16,"July")] [a,b] [] [(a,[[(16,"July")]]),(b,[[(16,"July")]])] []
Albert connaissait le mois, et en a déduit la bonne date sans hésiter après l'annonce de Bernard. Pour nous, ce ne peut donc pas être en août, où il y avait deux options.

Et voilà que

tout le monde s'accorde,

Cheryl est née le 16 juillet.

* Petite note : Albert, de toute façon, ne peut pas connaître la date d'anniversaire de Cheryl à la première étape, et Bernard s'en doute : pas besoin de le mettre au courant. On s'est donc permis de restreindre d'emblée la première annonce d'Albert à sa deuxième composante, la seule qui soit informative, et de passer sous silence la partie superflue entre crochets.

Le jeu des as et des huits (avec Enzo et Julien)

On distribue huit cartes à jouer (quatre huit, quatre as) à trois joueurs. Neuf parties sont menées sans que tou.te.s soient sûr.e.s de maîtriser le modèle. Elles s'arrêtent dès qu'un joueur parvient à déterminer les cartes qu'iel a dans la main à partir de celles des deux autres et de leurs annonces, données dans un ordre variable. Au bout de deux tours, on cesse le jeu de toute façon pour noter les donnes : les annonces d'un joueur se notent ✓ s'iel a trouvé, × s'iel a tort, ▯ s'iel ne sait pas. Entre crochets, les cartes effectivement distribuées à chacun. Le chiffre indique l'ordre de parole.

```
(1)    {[81](3,✓), [88](1,▯), [81](2,▯)}                (5)    {[18](1,▯), [11](2,▯), [88](3,✓)}
(2)    {[11](2,×), [81](),    [81](1,▯)}                (6)    {[18](1,▯), [18](2,▯), [18](3,✓)}
(3)    {[88](1,▯), [88](2,▯), [11](3,✓)}                (7)    {[18](2,▯), [18](3,▯), [18](1,▯)(4,×)}
(4)    {[18](2,▯), [88](3,✓), [11](1,▯)}                (8)    {[18](1,▯)(4,✓), [18](2,▯), [18](3,▯)}
                        (9) {[18](3,▯)(6,×), [18](1,▯)(4,▯), [18](2,▯)(5,▯)}
```

Sur la partie n°9, il aurait été difficile pour quiconque de déterminer ses cartes en deux tours, étant donné qu'il n'y a aucune paire. La partie n°8 est probablement un cas de bluff criant. Une partie qui respecte la structure de Kripke (un peu difficile à reproduire ici) de façon évidente est la troisième : voir deux paires permet de déduire que l'on en a une (du signe opposé).

conclusion

Ce septième TME a introduit une nouvelle façon de considérer la logique S5 que l'on connaissait déjà. Les fonctionnalités de Haskell et en particulier de DEMO-S5 ont permis de dérouler des raisonnements un peu plus complexes, et de résoudre des énigmes épistémiques du type de l'anniversaire de Cheryl, autrement assez délicats à gérer.

N'ayant pas eu à définir des modèles par nous-mêmes, nous avons constaté que l'intérêt majeur de ce TME n'était pas vraiment la découverte d'une nouvelle syntaxe (quoique) ; la principale nouveauté est la possibilité de mettre à jour un modèle préexistant en simulant des annonces publiques (syntaxe `upd_pa` de DEMO-S5). Un ajout d'information est donc représenté par une réduction des potentialités : moins de mondes possibles pour chaque agent, donc moins d'hésitations, jusqu'à ce que tou.te.s acquièrent une certitude : *voici le monde où je me trouve.* Le processus étant une source assez fiable d'erreurs de néophytes, l'automatiser nous a permis de nous concentrer sur l'essentiel, et de mieux comprendre le raisonnement filé en TD.

A ce propos, on notera que réduire le modèle à un seul monde revient à faire de ce qui s'y sait une connaissance à la fois partagée, commune et distribuée : tous les agents savent qu'ils sont dans la situation représentée, et tous savent que tous le savent, et le sauraient encore s'ils pouvaient en discuter.