

Appendix A Jobs Puzzle in APC_{LP} with Inconsistency Injections

We now present a complete APC_{LP} encoding of Jobs Puzzle and highlight the principles, introduced in Section 6, used in the encoding. We also show several cases of inconsistency injection and discuss the consequences. The English sentences are based on the CNL representation of Jobs Puzzle from Section 3 in (Schwitter 2013) where “Steve” is changed to “Robin” for the sake of an example (because Robin can be both a male and a female name).

- 1 Roberta is a person. Thelma is a person. Robin is a person. Pete is a person.
 $person(roberta) : \mathbf{t}. \quad person(thelma) : \mathbf{t}. \quad person(robin) : \mathbf{t}. \quad person(pete) : \mathbf{t}.$
- 2 Roberta is a female. Thelma is a female.
 $female(roberta) : \mathbf{t}. \quad female(thelma) : \mathbf{t}.$
- 3 Robin is male. Pete is male.
 $male(robin) : \mathbf{t}. \quad male(pete) : \mathbf{t}.$

Sentence 4 is encoded based on Principle 3, which treats *male* and *female* as polar facts.

- 4 Exclude that a person is male and that the person is female.
 $male(X) : \mathbf{t} \vee female(X) : \mathbf{t} \leftarrow person(X) : \mathbf{t}.$
 $male(X) : \mathbf{f} \vee female(X) : \mathbf{f} \leftarrow person(X) : \mathbf{t}.$
 $female(X) : \top \leftarrow person(X) : \mathbf{t} \wedge male(X) : \top.$
 $male(X) : \top \leftarrow person(X) : \mathbf{t} \wedge female(X) : \top.$

We encode sentences 5 and 6 using Principle 6, which constrains the cardinality of $hold(X, Y)$. This will cause second rule in Sentence 5 to be repeated as part of encoding of Sentence 6, so we omit the duplicate.

- 5 If there is a job then exactly one person holds that job.
 $1 \{hold(X, Y) : \mathbf{t} \mid if \quad person(X) : \mathbf{t}\} 1 \leftarrow job(Y) : \mathbf{t}.$
 $hold(X, Y) : \mathbf{f} \leftarrow person(X) : \mathbf{t} \wedge job(Y) : \mathbf{t} \wedge \neg hold(X, Y) : \mathbf{t}.$
- 6 If there is a person then the person holds exactly two jobs.
 $2 \{hold(X, Y) : \mathbf{t} \mid if \quad job(Y) : \mathbf{t}\} 2 \leftarrow person(X) : \mathbf{t}.$

Encoding of the following facts is straightforward:

- 7 Chef is a job. Guard is a job. Nurse is a job. Operator is a job. Police is a job. Teacher is a job. Actor is a job. Boxer is a job.
 $job(chef) : \mathbf{t}. \quad job(guard) : \mathbf{t}. \quad job(nurse) : \mathbf{t}. \quad job(operator) : \mathbf{t}.$
 $job(police) : \mathbf{t}. \quad job(teacher) : \mathbf{t}. \quad job(actor) : \mathbf{t}. \quad job(boxer) : \mathbf{t}.$

Sentences 8-13 are encoded based on Principles 1 and 2, where contrapositive inference and propagation of inconsistency are allowed for some literals but not others. Notice that it is undesirable to allow propagation of inconsistency from *person*-facts and *job*-facts, since it is unreasonable to conclude that somebody is, say, a male while being unsure that this somebody is a person. Ditto about the jobs. Contrapositive reasoning (say, from non-male to non-person) is also inappropriate here because we have higher confidence in someone being a person. So, we use ontological implication $\leftarrow$ in the next group of rules.

- 8 If a person holds a job as a nurse then that person is a male.
 $(male(X) : \mathbf{t} \leftarrow hold(X, nurse) : \mathbf{t}) \leftarrow person(X) : \mathbf{t} \wedge job(nurse) : \mathbf{t} \wedge \neg person(X) : \top \wedge \neg job(nurse) : \top.$

- 9 If a person holds a job as an actor then that person is a male.

$$(male(X) : \mathbf{t} \Leftarrow hold(X, actor) : \mathbf{t}) \leftarrow person(X) : \mathbf{t} \wedge job(actor) : \mathbf{t} \wedge \neg person(X) : \top \wedge \neg job(actor) : \top.$$

- 10 If a first person holds a job as a chef and a second person holds a job as a telephone operator then the second person is a husband of the first person.

$$(husband(Y, X) : \mathbf{t} \Leftarrow hold(X, chef) : \mathbf{t} \wedge hold(Y, operator) : \mathbf{t}) \leftarrow person(X) : \mathbf{t} \wedge job(chef) : \mathbf{t} \wedge person(Y) : \mathbf{t} \wedge job(operator) : \mathbf{t} \wedge \neg person(X) : \top \wedge \neg job(chef) : \top \wedge \neg person(Y) : \top \wedge \neg job(operator) : \top.$$

- 11 If a first person is a husband of a second person then the first person is male.

$$(male(X) : \mathbf{t} \Leftarrow husband(X, Y) : \mathbf{t}) \leftarrow person(X) : \mathbf{t} \wedge person(Y) : \mathbf{t} \wedge \neg person(X) : \top \wedge \neg person(Y) : \top.$$

- 12 If a first person is a husband of a second person then the second person is female.

$$(female(Y) : \mathbf{t} \Leftarrow husband(X, Y) : \mathbf{t}) \leftarrow person(X) : \mathbf{t} \wedge person(Y) : \mathbf{t} \wedge \neg person(X) : \top \wedge \neg person(Y) : \top.$$

- 13 Exclude that Roberta holds a job as boxer.

$$hold(roberta, boxer) : \mathbf{f} \leftarrow job(boxer) : \mathbf{t} \wedge \neg job(boxer) : \top.$$

Encoding of the following fact is straightforward.

- 14 Exclude that Pete is educated.

$$educated(pete) : \mathbf{f}.$$

Sentences 15-20 are also encoded based on Principles 1 and 2.

- 15 If a person holds a job as nurse then the person is educated.

$$(educated(X) : \mathbf{t} \Leftarrow hold(X, nurse) : \mathbf{t}) \leftarrow person(X) : \mathbf{t} \wedge job(nurse) : \mathbf{t} \wedge \neg person(X) : \top \wedge \neg job(nurse) : \top.$$

- 16 If a person holds a job as a police officer then that person is educated.

$$(educated(X) : \mathbf{t} \Leftarrow hold(X, police) : \mathbf{t}) \leftarrow person(X) : \mathbf{t} \wedge job(police) : \mathbf{t} \wedge \neg person(X) : \top \wedge \neg job(police) : \top.$$

- 17 If a person holds a job as a teacher then the person is educated.

$$(educated(X) : \mathbf{t} \Leftarrow hold(X, teacher) : \mathbf{t}) \leftarrow person(X) : \mathbf{t} \wedge job(teacher) : \mathbf{t} \wedge \neg person(X) : \top \wedge \neg job(teacher) : \top.$$

- 18 Exclude that Roberta holds a job as a chef.

$$hold(roberta, chef) : \mathbf{f} \leftarrow job(chef) : \mathbf{t} \wedge \neg job(chef) : \top.$$

- 19 Exclude that Roberta holds a job as a police officer.

$$hold(roberta, police) : \mathbf{f} \leftarrow job(police) : \mathbf{t} \wedge \neg job(police) : \top.$$

- 20 Exclude that a person holds a job as a chef and that the same person holds a job as a police officer.

$$hold(X, chef) : \mathbf{f} \vee hold(X, police) : \mathbf{f} \leftarrow person(X) : \mathbf{t} \wedge job(chef) : \mathbf{t} \wedge job(police) : \mathbf{t} \wedge \neg person(X) : \top \wedge \neg job(chef) : \top \wedge \neg job(police) : \top.$$

Next we define the consistency preference relation $\prec_{\mathcal{S}}$, where $\mathcal{S} = (s_1, s_2, s_3, \mathcal{B}_{\top})$, which implements Principle 4. Here s_1 says that we hold greater confidence in the information about someone being a person and something being a job than in any other kind of information in the puzzle. That is, these facts are least likely to be inconsistent. Next, s_2 says that we are very likely to believe that Pete is a male name and Thelma and Roberta are female names. We are not sure about Robin, so s/he is left out in s_2 . The set s_3 says that next we are likely to believe the

information on who holds which jobs. The last component in $\mathcal{S}$, $\mathcal{B}_\top$, is the usual default that prefers the most e-consistent models.

$$s_1 = \{ \text{person}(\text{roberta}) : \top, \text{person}(\text{thelma}) : \top, \text{person}(\text{robin}) : \top, \text{person}(\text{pete}) : \top, \\ \text{job}(\text{chef}) : \top, \text{job}(\text{guard}) : \top, \text{job}(\text{nurse}) : \top, \text{job}(\text{operator}) : \top, \\ \text{job}(\text{police}) : \top, \text{job}(\text{teacher}) : \top, \text{job}(\text{actor}) : \top, \text{job}(\text{boxer}) : \top \}$$

$$s_2 = \{ \text{male}(\text{pete}) : \top, \text{female}(\text{thelma}) : \top, \text{female}(\text{roberta}) : \top \}$$

$$s_3 = \{ \text{hold}(\text{pete}, \text{chef}) : \top, \text{hold}(\text{pete}, \text{guard}) : \top, \text{hold}(\text{pete}, \text{nurse}) : \top, \text{hold}(\text{pete}, \\ \text{operator}) : \top, \text{hold}(\text{pete}, \text{police}) : \top, \text{hold}(\text{pete}, \text{teacher}) : \top, \text{hold}(\text{pete}, \\ \text{actor}) : \top, \text{hold}(\text{pete}, \text{boxer}) : \top, \dots \}$$

It is interesting to note that if s_3 is not included then, in some cases, there might be too many possibilities to solve the puzzle by allowing *hold*-predicates to be inconsistent. This is because many rules and constraints in the puzzle use *hold* as a premise. So, without minimizing the possibility of inconsistency in *hold* those rules and constraints become vacuously true, leading to an explosion of the possible worlds.

The APC_{LP} encoding generates one most consistency preferred model where the information concerning *hold*/2 is

$$\begin{array}{lll} \text{hold}(\text{pete}, \text{actor}) : \mathbf{t} & \text{hold}(\text{pete}, \text{operator}) : \mathbf{t} & \text{hold}(\text{robin}, \text{police}) : \mathbf{t} \\ \text{hold}(\text{robin}, \text{nurse}) : \mathbf{t} & \text{hold}(\text{thelma}, \text{chef}) : \mathbf{t} & \text{hold}(\text{thelma}, \text{boxer}) : \mathbf{t} \\ \text{hold}(\text{roberta}, \text{guard}) : \mathbf{t} & \text{hold}(\text{roberta}, \text{teacher}) : \mathbf{t} & \end{array}$$

Not surprisingly, this solution matches other approaches because so far we have not injected inconsistency (and so, for example, the preferences s_1 , s_2 , and s_3 do not matter here).

Next, we illustrate five cases of injection of inconsistency into the puzzle. Since complete models tend to be rather large, we show only *hold*/2 and $\top$ -predicates.

21 Thelma is an actor — Variation 1

$$\text{hold}(\text{thelma}, \text{actor}) : \mathbf{t}.$$

This makes Thelma's job assignment as an actor inconsistent and we get this model:

$$m = \{ \text{hold}(\text{pete}, \text{boxer}) : \mathbf{t} \quad \text{hold}(\text{pete}, \text{operator}) : \mathbf{t} \quad \text{hold}(\text{robin}, \text{police}) : \mathbf{t} \\ \text{hold}(\text{robin}, \text{nurse}) : \mathbf{t} \quad \text{hold}(\text{thelma}, \text{actor}) : \top \quad \text{hold}(\text{thelma}, \text{chef}) : \mathbf{t} \\ \text{hold}(\text{roberta}, \text{teacher}) : \mathbf{t} \quad \text{hold}(\text{roberta}, \text{guard}) : \mathbf{t} \}$$

Indeed, the original puzzle implies that Thelma is *not* an actor. Given that $\text{hold}(\text{thelma}, \text{actor}) : \mathbf{t}$ is true, Sentence 9 will sanction two possibilities: one where $\text{male}(\text{thelma}) : \mathbf{t}$ is true and the other where $\text{hold}(\text{thelma}, \text{actor}) : \top$ is true. In the first case, Sentence 4 will force $\text{male}(\text{thelma}) : \top$ and $\text{female}(\text{thelma}) : \top$ into the model. In the second case, we will have $\text{hold}(\text{thelma}, \text{actor}) : \top$ in the model. Given that we have high confidence in Thelma's gender (preference s_2), we hold her gender less likely to be inconsistent. Thus, the first case gets eliminated.

The next variation assumes that Robin is a female (instead of Thelma being an actor in Variation 1).

22 Robin is female — Variation 2

$$\text{female}(\text{robin}) : \mathbf{t}.$$

There are two models:

$$\begin{aligned}
m_1 = & \{ \text{hold}(\text{pete}, \text{actor}) : \mathbf{t} \text{ hold}(\text{pete}, \text{operator}) : \mathbf{t} \text{ hold}(\text{robin}, \text{police}) : \mathbf{t} \\
& \text{hold}(\text{robin}, \text{nurse}) : \mathbf{t} \text{ hold}(\text{roberta}, \text{teacher}) : \mathbf{t} \text{ hold}(\text{thelma}, \text{boxer}) : \mathbf{t} \\
& \text{hold}(\text{roberta}, \text{guard}) : \mathbf{t} \text{ hold}(\text{thelma}, \text{chef}) : \mathbf{t} \text{ husband}(\text{pete}, \text{thelma}) : \mathbf{t} \\
& \text{male}(\text{robin}) : \top \text{ female}(\text{robin}) : \top \} \\
m_2 = & \{ \text{hold}(\text{pete}, \text{actor}) : \mathbf{t} \text{ hold}(\text{pete}, \text{operator}) : \mathbf{t} \text{ hold}(\text{robin}, \text{nurse}) : \mathbf{t} \\
& \text{hold}(\text{robin}, \text{chef}) : \mathbf{t} \text{ hold}(\text{thelma}, \text{boxer}) : \mathbf{t} \text{ hold}(\text{thelma}, \text{police}) : \mathbf{t} \\
& \text{hold}(\text{roberta}, \text{guard}) : \mathbf{t} \text{ hold}(\text{roberta}, \text{teacher}) : \mathbf{t} \text{ husband}(\text{pete}, \text{robin}) : \mathbf{t} \\
& \text{female}(\text{robin}) : \top \text{ male}(\text{robin}) : \top \}
\end{aligned}$$

Sentences 3 and 4 imply inconsistency regarding Robin's gender. The first model is the same as in the original puzzle (as far as the job assignments go). In the second model, since Robin's gender is inconsistent (both male and female), it is compatible to make Robin a chef and Thelma a police officer. Therefore, we derive that Pete is Robin's husband instead of Thelma's.

The next variation of the original puzzle explicitly assumes that Robin is Thelma's husband.

23 Robin is a husband of Thelma — Variation 3

$$\text{husband}(\text{robin}, \text{thelma}) : \mathbf{t}.$$

Here we need to add more background knowledge about marital relations. For instance, that every person can marry or be married to at most one person. Together with Sentence 23, this will cause inconsistency because the original puzzle implies that Pete is Thelma's husband. In (Schwitter 2013), this implicit knowledge is not stated, so it will fail to detect inconsistency. If such background knowledge were added to Schwitter's formulation as constraints then there would be no models. The background knowledge we need is:

- 24 A person who is a male is a husband of exactly one other person, or that person is null.
- 25 A person who is a female has exactly one husband or that husband is null.
- 26 Exclude that person X is a husband of Y and person Z is a husband of X simultaneously.
- 27 If it is not derivable that person X is person Y's husband, then X is not Y's husband.

Sentences 24 – 26 are cardinality constraints and are encoded based on Principle 6. Sentence 27 says that the information about husbands is complete; it is encoded based on Principle 5. Also, we block propagation of inconsistency from male and female based on Principle 2.

$$\begin{aligned}
& 1 \{ \text{husband}(X, Y) : \mathbf{t} \text{ if } (\text{person}(Y) : \mathbf{t} \text{ or } Y = \text{null}) \} 1 \leftarrow \\
& \quad \text{person}(X) : \mathbf{t} \wedge \text{male}(X) : \mathbf{t} \wedge \neg \text{male}(X) : \top. \\
& 1 \{ \text{husband}(X, Y) : \mathbf{t} \text{ if } (\text{person}(X) : \mathbf{t} \text{ or } X = \text{null}) \} 1 \leftarrow \\
& \quad \text{person}(Y) : \mathbf{t} \wedge \text{female}(Y) : \mathbf{t} \wedge \neg \text{female}(Y) : \top. \\
& :- \text{husband}(X, Y) : \mathbf{t} \wedge \text{husband}(Z, X) : \mathbf{t} \wedge X \neq \text{null}. \\
& \text{husband}(X, Y) : \mathbf{f} \leftarrow \text{person}(X) : \mathbf{t} \wedge \text{person}(Y) : \mathbf{t} \wedge \neg \text{husband}(X, Y) : \mathbf{t}.
\end{aligned}$$

There are six models. When these models are projected on *hold*/2 (which constitutes the solution to the puzzle) and the $\top$ -predicates, we get three distinct sets:

$$\begin{aligned}
m_1 = & \{ \text{hold}(\text{pete}, \text{actor}) : \mathbf{t} \quad \text{hold}(\text{pete}, \text{police}) : \mathbf{t} \text{ hold}(\text{robin}, \text{operator}) : \mathbf{t} \\
& \text{hold}(\text{robin}, \text{nurse}) : \mathbf{t} \text{ hold}(\text{roberta}, \text{teacher}) : \mathbf{t} \text{ hold}(\text{roberta}, \text{guard}) : \mathbf{t} \\
& \text{hold}(\text{thelma}, \text{boxer}) : \mathbf{t} \text{ hold}(\text{thelma}, \text{chef}) : \mathbf{t} \text{ educated}(\text{pete}) : \top \} \\
m_2 = & \{ \text{hold}(\text{pete}, \text{actor}) : \mathbf{t} \text{ hold}(\text{pete}, \text{nurse}) : \mathbf{t} \text{ hold}(\text{robin}, \text{operator}) : \mathbf{t} \\
& \text{hold}(\text{robin}, \text{police}) : \mathbf{t} \text{ hold}(\text{roberta}, \text{teacher}) : \mathbf{t} \text{ hold}(\text{roberta}, \text{guard}) : \mathbf{t}
\end{aligned}$$

$$\begin{aligned}
& hold(thelma, boxer) : \mathbf{t} \quad hold(thelma, chef) : \mathbf{t} \quad educated(pete) : \top \\
m_3 = & \{ hold(pete, police) : \mathbf{t} \quad hold(pete, nurse) : \mathbf{t} \quad hold(robin, operator) : \mathbf{t} \\
& hold(robin, actor) : \mathbf{t} \quad hold(roberta, teacher) : \mathbf{t} \quad hold(roberta, guard) : \mathbf{t} \\
& hold(thelma, boxer) : \mathbf{t} \quad hold(thelma, chef) : \mathbf{t} \quad educated(pete) : \top \}
\end{aligned}$$

The puzzle originally implied that Pete is Thelma's husband. Since we now explicitly stated that Robin is Thelma's husband, Sentences 25 and 27 will force $husband(pete, thelma) : \mathbf{f}$ to hold. By Sentence 10, $hold(pete, operator) : \mathbf{t}$ and $hold(Thelma, chef) : \mathbf{t}$ cannot hold simultaneously, so many solutions with inconsistencies in them will be generated. Due to the consistency preference relations, the APC_{LP} encoding will prefer the models where $educated(pete) : \top$ holds.

Sentences 24 and 25 sanction two possibilities for the husband information in each of the above models. For instance, the model m_1 corresponds to two models out of the six models that we get; they differ only in their husband information. Given that Pete is not an operator and Roberta is not a chef, Pete is not necessarily Roberta's husband. Therefore, there are two cases: one where $husband(pete, roberta) : \mathbf{t}$ holds and the other where $husband(pete, null) : \mathbf{t}$ and $husband(null, roberta) : \mathbf{t}$ hold. Similar considerations apply to m_2 and m_3 .

The next variation applies the background knowledge about husbands from Variation 3 to Variations 1 and 2.

28 Thelma is an actor. — Variation 4.1 (modification of Variation 1)

$$hold(thelma, actor) : \mathbf{t}.$$

Sentences 24 – 27.

There are now two models, and their projections on *hold*- and $\top$ -predicates are compatible with the solution to Variation 1. The only difference between these two models is in *husband*-predicates, so we show only that part.

$$\begin{aligned}
m_1 = & \{ husband(pete, thelma) : \mathbf{t}, \quad husband(robin, roberta) : \mathbf{t} \} \\
m_2 = & \{ husband(pete, thelma) : \mathbf{t}, \quad husband(robin, null) : \mathbf{t}, \quad husband(null, roberta) : \mathbf{t} \}
\end{aligned}$$

29 Robin is a female. — Variation 4.2 (modification of Variation 2)

$$female(robin) : \mathbf{t}.$$

Sentences 24 – 27.

There are three models and their projections on *hold*- and $\top$ -predicates are compatible with Variation 2. The only difference with Variation 2 is in the *husband*-predicates, which we show:

$$\begin{aligned}
m_1 = & \{ husband(pete, thelma) : \mathbf{t}, \quad husband(robin, roberta) : \mathbf{t} \} \\
m_2 = & \{ husband(pete, thelma) : \mathbf{t}, \quad husband(robin, null) : \mathbf{t}, \quad husband(null, roberta) : \mathbf{t} \} \\
m_3 = & \{ husband(pete, robin) : \mathbf{t}, \quad husband(null, thelma) : \mathbf{t}, \quad husband(null, roberta) : \mathbf{t} \}
\end{aligned}$$

Here m_1 and m_2 correspond to the first model in Variation 2 and m_3 corresponds to the second model there.

Appendix B Zebra Puzzle in APC_{LP} with Inconsistency Injections

We now present a complete APC_{LP} encoding of the original Zebra Puzzle as is described in Wikipedia.⁷ A slightly different version of the puzzle appears in TPTP.⁸ The encoding highlights the principles introduced in Section 6 and we also discuss several cases of inconsistency injection.

- (a) There are five houses.
- (b) The Englishman lives in the red house.
- (c) The Spaniard owns the dog.
- (d) Coffee is drunk in the green house.
- (e) The Ukrainian drinks tea.
- (f) The green house is immediately to the right of the ivory house.
- (g) The Old Gold smoker owns snails.
- (h) Kools are smoked in the yellow house.
- (i) Milk is drunk in the middle house.
- (j) The Norwegian lives in the first house.
- (k) The man who smokes Chesterfields lives in the house next to the man with the fox.
- (l) Kools are smoked in the house next to the house where the horse is kept.
- (m) The Lucky Strike smoker drinks orange juice.
- (n) The Japanese smokes Parliaments.
- (o) The Norwegian lives next to the blue house.

Now, who drinks water? Who owns the zebra?

In the interest of clarity, it must be added that each of the five houses is painted a different color, and their inhabitants are of different national extractions, own different pets, drink different beverages and smoke different brands of American cigarettes [sic]. One other thing: in statement 6, right means your right.

— Life International, December 17, 1962

Zebra Puzzle implies some background knowledge, which we must add. First, there is a one-to-one correspondence between houses and colors (and persons, drinks, cigarettes, and pets). Second, we assume that the first house stands on the extreme left and the fifth house stands on the extreme right. The first and the fifth house are not next to each other. A house is on the right of another house if the latter is to the left of the former. Right-to and left-to imply next-to and next-to is a symmetric relation. Next, we list the facts and the rules that encode the puzzle. For clarity statement is related to an appropriate English statement from the puzzle or from the implicit information mentioned above.

Sentences 1-5 provide the *house*-, *color*-, *person*-, *cigarette*-, and *pet*-facts.

- 1 There are five houses: #1, #2, #3, #4, and #5.
 $house(1) : \mathbf{t.} \quad house(2) : \mathbf{t.} \quad house(3) : \mathbf{t.} \quad house(4) : \mathbf{t.} \quad house(5) : \mathbf{t.}$
- 2 There are five colors: yellow, blue, red, ivory, and green.
 $color(yellow) : \mathbf{t.} \quad color(blue) : \mathbf{t.} \quad color(red) : \mathbf{t.} \quad color(ivory) : \mathbf{t.} \quad color(green) : \mathbf{t.}$
- 3 There are five people: Norwegian, Ukrainian, Englishman, Spaniard, and Japanese.

⁷ https://en.wikipedia.org/wiki/Zebra_Puzzle

⁸ <http://www.cs.miami.edu/~tptp/cgi-bin/SeeTPTP?Category=Problems&Domain=PUZ&File=PUZ010-1.p>

$person(norwegian) : \mathbf{t}$. $person(ukrainian) : \mathbf{t}$. $person(englishman) : \mathbf{t}$.
 $person(spaniard) : \mathbf{t}$. $person(japanese) : \mathbf{t}$.

- 4 There are five drinks: water, tea, milk, orange juice, and coffee.

$drink(water) : \mathbf{t}$. $drink(tea) : \mathbf{t}$. $drink(milk) : \mathbf{t}$. $drink(orange_juice) : \mathbf{t}$.
 $drink(coffee) : \mathbf{t}$.

- 5 There are five cigarettes: Kools, Chesterfield, Old Gold, Luck Strike, and Parliament.

$cigarette(kools) : \mathbf{t}$. $cigarette(chesterfield) : \mathbf{t}$. $cigarette(old_gold) : \mathbf{t}$.
 $cigarette(lucky_strike) : \mathbf{t}$. $cigarette(parliament) : \mathbf{t}$.

- 6 There are five pets: fox, horse, snails, dog, and zebra.

$pet(fox) : \mathbf{t}$. $pet(horse) : \mathbf{t}$. $pet(snails) : \mathbf{t}$. $pet(dog) : \mathbf{t}$. $pet(zebra) : \mathbf{t}$.

Sentences 7-16 describe the implicit knowledge about the one-to-one correspondence between houses, colors, persons, drinks, cigarettes, and pets. They are encoded as the cardinality constraints on the predicates $house_color(H, C)$, $house_nationality(H, N)$, $house_drink(H, D)$, $house_smoke(H, S)$, and $house_pet(H, P)$. We encode Sentences 7 - 8 using Principles 5 and 6. This will cause the second rule in Sentence 8 to be repeated as part of the encoding of Sentence 7, so we omit the duplicate. The rest of the sentences follow the same idea.

- 7 Each house has exactly one color.

$1\{house_color(H, C) : \mathbf{t} \text{ if } color(C) : \mathbf{t}\}1 \leftarrow house(H) : \mathbf{t}$.
 $house_color(H, C) : \mathbf{f} \leftarrow house(H) : \mathbf{t} \wedge color(C) : \mathbf{t} \wedge \neg house_color(H, C) : \mathbf{t}$.

- 8 Each color is for exactly one house.

$1\{house_color(H, C) : \mathbf{t} \text{ if } house(H) : \mathbf{t}\}1 \leftarrow color(C) : \mathbf{t}$.

- 9 Each house is home for exactly one person.

$1\{house_nationality(H, N) : \mathbf{t} \text{ if } person(N) : \mathbf{t}\}1 \leftarrow house(H) : \mathbf{t}$.
 $house_nationality(H, N) : \mathbf{f} \leftarrow house(H) : \mathbf{t} \wedge person(N) : \mathbf{t} \wedge$
 $\neg house_nationality(H, N) : \mathbf{t}$.

- 10 Each person lives in exactly one house.

$1\{house_nationality(H, N) : \mathbf{t} \text{ if } house(H) : \mathbf{t}\}1 \leftarrow person(N) : \mathbf{t}$.

- 11 Each house has exactly one favorite drink.

$1\{house_drink(H, D) : \mathbf{t} \text{ if } drink(D) : \mathbf{t}\}1 \leftarrow house(H) : \mathbf{t}$.
 $house_drink(H, D) : \mathbf{f} \leftarrow house(H) : \mathbf{t} \wedge drink(D) : \mathbf{t} \wedge \neg house_drink(H, D) : \mathbf{t}$.

- 12 Each drink is drunk in exactly one house.

$1\{house_drink(H, D) : \mathbf{t} \text{ if } house(H) : \mathbf{t}\}1 \leftarrow drink(D) : \mathbf{t}$.

- 13 Each house has exactly one brand of cigarettes.

$1\{house_smoke(H, S) : \mathbf{t} \text{ if } cigarette(S) : \mathbf{t}\}1 \leftarrow house(H) : \mathbf{t}$.
 $house_smoke(H, S) : \mathbf{f} \leftarrow house(H) : \mathbf{t} \wedge cigarette(S) : \mathbf{t} \wedge \neg house_smoke(H, S) : \mathbf{t}$.

- 14 Each brand of cigarettes is smoked in exactly one house.

$1\{house_smoke(H, S) : \mathbf{t} \text{ if } house(H) : \mathbf{t}\}1 \leftarrow cigarette(S) : \mathbf{t}$.

- 15 Each house has exactly one pet.

$1\{house_pet(H, P) : \mathbf{t} \text{ if } pet(P) : \mathbf{t}\}1 \leftarrow house(H) : \mathbf{t}$.
 $house_pet(H, P) : \mathbf{f} \leftarrow house(H) : \mathbf{t} \wedge pet(P) : \mathbf{t} \wedge \neg house_pet(H, P) : \mathbf{t}$.

- 16 Each pet is kept in exactly one house.

$1\{house_pet(H, P) : \mathbf{t} \text{ if } house(H) : \mathbf{t}\}1 \leftarrow pet(P) : \mathbf{t}$.

Sentences 17-20 correspond to Sentences (b)-(e) in the original puzzle. The encoding is based on Principles 1 and 2, where contrapositive inference and propagation of inconsistency are allowed for some literals but not others. Notice that it is undesirable to allow propagation of inconsistency from *house*-facts, since it is unreasonable to conclude that somebody lives in a house

while being unsure that something is a house. Contrapositive reasoning is also inappropriate here. For instance, if there is uncertainty about the color of a house, it is unreasonable to conclude that something is not a house because one is much more likely to discern a house than its color. So, in the next group of rules, we separate $house(H)$ from color and other facts using use ontological implication $\leftarrow$.

- 17 If the Englishman lives in a house then the color of the house is red.
 $(house_color(H, red) : \mathbf{t} \leftarrow house_nationality(H, englishman) : \mathbf{t})$
 $\leftarrow house(H) : \mathbf{t} \wedge \neg house(H) : \top.$
- 18 If the Spaniard lives in a house then dog is kept in the house.
 $(house_pet(H, dog) : \mathbf{t} \leftarrow house_nationality(H, spaniard) : \mathbf{t})$
 $\leftarrow house(H) : \mathbf{t} \wedge \neg house(H) : \top.$
- 19 If the color of a house is green then coffee is drunk in the house.
 $(house_drink(H, coffee) : \mathbf{t} \leftarrow house_color(H, green) : \mathbf{t})$
 $\leftarrow house(H) : \mathbf{t} \wedge \neg house(H) : \top.$
- 20 If the Ukrainian lives in a house then tea is drunk in the house.
 $(house_drink(H, tea) : \mathbf{t} \leftarrow house_nationality(H, ukrainian) : \mathbf{t})$
 $\leftarrow house(H) : \mathbf{t} \wedge \neg house(H) : \top.$

Sentences 21-22 define the implicit knowledge of the $right(X, Y)$ relation. We encode each sentence based on Principle 2.

- 21 A house numbered X is to the right of another house numbered Y if $X - 1 = Y$.
 $right(X, Y) : \mathbf{t} \leftarrow house(X) : \mathbf{t} \wedge house(Y) : \mathbf{t} \wedge X - 1 = Y.$
 $\neg house(X) : \top \wedge \neg house(Y) : \top.$
- 22 A house numbered X is not to right of another house numbered Y if $X - 1 \neq Y$.
 $right(X, Y) : \mathbf{f} \leftarrow house(X) : \mathbf{t} \wedge house(Y) : \mathbf{t} \wedge X - 1 \neq Y.$
 $\neg house(X) : \top \wedge \neg house(Y) : \top.$

Sentences 23-25 correspond to Sentence (f). The encoding of Sentences 23-24 is straightforward. The encoding of Sentence 25 is based on Principles 1 and 2.

- 23 The color of the first house is not green.
 $house_color(1, green) : \mathbf{f}.$
- 24 The color of the fifth house is not ivory.
 $house_color(5, ivory) : \mathbf{f}.$
- 25 If a house is to the right of another house and the color of the former house is green then the color of the latter house is ivory.
 $(house_color(Left, ivory) : \mathbf{t} \leftarrow house_color(Right, green) : \mathbf{t})$
 $\leftarrow house(Left) : \mathbf{t} \wedge house(Right) : \mathbf{t} \wedge right(Right, Left) : \mathbf{t} \wedge$
 $\neg house(Left) : \top \wedge \neg house(Right) : \top \wedge \neg right(Right, Left) : \top.$

Sentences 26-27 below correspond to Sentences (g)-(h). The encoding is based on Principles 1 and 2.

- 26 If Old Gold is smoked in a house then snails are kept in the house.
 $(house_pet(H, snails) : \mathbf{t} \leftarrow house_smoke(H, old_gold) : \mathbf{t})$
 $\leftarrow house(H) : \mathbf{t} \wedge \neg house(H) : \top.$
- 27 If Kools is smoked in a house then the color of the house is yellow.

$$(house_smoke(H, kools) : \mathbf{t} \triangleleft \sim house_color(H, yellow) : \mathbf{t}) \\ \leftarrow house(H) : \mathbf{t} \wedge \neg house(H) : \top.$$

Sentences 28-29 below correspond to Sentences (i)-(j). The encoding is straightforward.

28 Milk is drunk in the middle house.

$$house_drink(3, milk) : \mathbf{t}.$$

29 The Norwegian lives in the first house.

$$house_nationality(1, norwegian) : \mathbf{t}.$$

Sentences 30-31 state the implicit knowledge of the $next(X, Y)$ relation. We encode each sentence based on Principle 2.

30 A house is next to another house if their house numbers differ by 1

$$next(X, Y) : \mathbf{t} \leftarrow house(X) : \mathbf{t} \wedge house(Y) : \mathbf{t} \wedge |X - Y| = 1. \\ \neg house(X) : \top \wedge \neg house(Y) : \top.$$

31 A house is not next to another house if their house numbers do not differ by 1

$$next(X, Y) : \mathbf{f} \leftarrow house(X) : \mathbf{t} \wedge house(Y) : \mathbf{t} \wedge |X - Y| \neq 1. \\ \neg house(X) : \top \wedge \neg house(Y) : \top.$$

Sentences 32-34 correspond to Sentence (k). Sentences 32-33 are encoded based on Principle 1. Sentence 34 is encoded based on Principles 1 and 2.

32 If Chesterfield is smoked in the first house then fox is kept in the second house.

$$house_pet(2, fox) : \mathbf{t} \triangleleft \sim house_smoke(1, chesterfield) : \mathbf{t}.$$

33 If Chesterfield is smoked in the fifth house then fox is kept in the fourth house.

$$house_pet(4, fox) : \mathbf{t} \triangleleft \sim house_smoke(5, chesterfield) : \mathbf{t}.$$

34 If a house is next to another house and Chesterfield is smoked in the former house then fox is kept in the latter house.

$$(house_pet(H1, fox) : \mathbf{t} \vee house_pet(H3, fox) : \mathbf{t} \triangleleft \sim house_smoke(H2, chesterfield) : \mathbf{t}) \\ \leftarrow house(H1) : \mathbf{t} \wedge house(H2) : \mathbf{t} \wedge house(H3) : \mathbf{t} \wedge \\ H1 \neq H3 \wedge next(H1, H2) : \mathbf{t} \wedge next(H2, H3) : \mathbf{t} \wedge \\ \neg house(H1) : \top \wedge \neg house(H2) : \top \wedge \neg house(H3) : \mathbf{t} : \top \wedge \\ \neg next(H1, H2) : \mathbf{t} : \top \wedge \neg next(H2, H3) : \top.$$

Sentences 35-37 correspond to Sentence (l). The encoding follows the same idea as Sentences 32-34.

35 If Kools are smoked in the first house then horse is kept in the second house

$$house_pet(2, horse) : \mathbf{t} \triangleleft \sim house_smoke(1, kools) : \mathbf{t}.$$

36 If Kools are smoked in the fifth house then horse is kept in the fourth house.

$$house_pet(4, horse) : \mathbf{t} \triangleleft \sim house_smoke(5, kools) : \mathbf{t}.$$

37 If a house is next to another house and Kools are smoked in the former house then horse is kept in the latter house.

$$(house_pet(H1, horse) : \mathbf{t} \vee house_pet(H3, horse) : \mathbf{t} \triangleleft \sim house_smoke(H2, kools) : \mathbf{t}) \\ \leftarrow house(H1) : \mathbf{t} \wedge house(H2) : \mathbf{t} \wedge house(H3) : \mathbf{t} \wedge \\ H1 \neq H3 \wedge next(H1, H2) : \mathbf{t} \wedge next(H2, H3) : \mathbf{t} \wedge \\ \neg house(H1) : \top \wedge \neg house(H2) : \top \wedge \neg house(H3) : \mathbf{t} : \top \wedge \\ \neg next(H1, H2) : \mathbf{t} : \top \wedge \neg next(H2, H3) : \top.$$

Sentences 38-39 below correspond to Sentences (m)-(n). The encoding is based on Principles 1 and 2.

- 38 If Lucky Strike is smoked in a house then orange juice is drunk in that house.
 $(house_drink(H, orange_juice) : \mathbf{t} \Leftarrow house_smoke(H, lucky_strike) : \mathbf{t})$
 $\leftarrow house(H) : \mathbf{t} \wedge \neg house(H) : \top.$
- 39 If the Japanese lives in a house then parliament is smoked in that house.
 $(house_smoke(H, parliament) : \mathbf{t} \Leftarrow house_nationality(H, japanese) : \mathbf{t})$
 $\leftarrow house(H) : \mathbf{t} \wedge \neg house(H) : \top.$

Sentences 40-42 correspond to Sentence (o). The encoding follows the same idea as Sentences 32-34.

- 40 If the Norwegian lives in the first house then the color of the second house is blue.
 $house_color(2, blue) : \mathbf{t} \Leftarrow house_nationality(1, norwegian) : \mathbf{t}.$
- 41 If the Norwegian lives in the fifth house then the color of the fourth house is blue.
 $house_color(4, blue) : \mathbf{t} \Leftarrow house_nationality(5, norwegian) : \mathbf{t}.$
- 42 If a house is next to another house and the Norwegian lives in the former house then the color of the latter house is blue.
 $(house_color(H1, blue) : \mathbf{t} \vee house_color(H3, blue) : \mathbf{t})$
 $\Leftarrow house_nationality(H2, norwegian) : \mathbf{t})$
 $\leftarrow house(H1) : \mathbf{t} \wedge house(H2) : \mathbf{t} \wedge house(H3) : \mathbf{t} \wedge$
 $H1 \neq H3 \wedge next(H1, H2) : \mathbf{t} \wedge next(H2, H3) : \mathbf{t} \wedge$
 $\neg house(H1) : \top \wedge \neg house(H2) : \top \wedge \neg house(H3) : \mathbf{t} : \top \wedge$
 $\neg next(H1, H2) : \mathbf{t} : \top \wedge \neg next(H2, H3) : \top.$

Next we define the consistency preference relation $<_{\mathcal{S}}$, where $\mathcal{S} = (s_1, s_2, \mathcal{B}_{\top})$, which implements Principle 4. Here s_1 says that we hold greater confidence in the information about the *house*-, *color*-, *person*-, *cigarette*-, and *pet*-, *next*, *right*-facts. We remind that “greater confidence” here means that these facts are least likely to be inconsistent. The preference s_2 says that next we are likely to be confident in the information given by the *house_color*(H, C), *house_nationality*(H, N), *house_drink*(H, D), *house_smoke*(H, C), *house_pet*(H, P) facts. The last component in $\mathcal{S}$, $\mathcal{B}_{\top}$, is the usual default that gives preference to the most e-consistent models.

$$s_1 = \{house(1) : \top, \dots, house(5) : \top, color(yellow) : \top, \dots, color(green) : \top, \\ person(norwegian) : \top, \dots, person(japanese) : \top, drink(water) : \top, \dots, \\ drink(coffe) : \top, cigarette(kools) : \top, \dots, cigarette(parliament) : \top, \\ pet(fox) : \top, \dots, pet(zebra) : \top\}.$$

$$s_2 = \{house_color(1, yellow) : \top, \dots, house_color(5, green) : \top, \\ house_nationality(1, norwegian) : \top, \dots, house_nationality(5, japanese) : \top, \\ house_drink(1, water) : \top, \dots, house_drink(5, coffe) : \top, \\ house_smoke(1, kools) : \top, \dots, house_smoke(5, parliament) : \top, \\ house_pet(1, fox) : \top, \dots, house_pet(5, zebra) : \top\}.$$

To better illustrate the result, we define a single predicate tuple/6 to represent the combined information of a house: its associated color, person, drink, cigarette, and pet.

$$tuple(H, C, N, D, S, P) : \mathbf{t} \leftarrow house_color(H, C) : \mathbf{t} \wedge house_nationality(H, N) : \mathbf{t} \wedge \\ house_drink(H, D) : \mathbf{t} \wedge house_smoke(H, S) : \mathbf{t} \wedge \\ house_pet(H, P) : \mathbf{t}.$$

There is a single most consistency preferred model:

tuple(1, yellow, norwegian, water, kools, fox) : **t**
tuple(2, blue, ukrainian, tea, chesterfield, horse) : **t**
tuple(3, red, englishman, milk, old_gold, snails) : **t**
tuple(4, ivory, spaniard, orange_juice, lucky_strike, dog) : **t**
tuple(5, green, japanese, coffee, parliament, zebra) : **t**

Not surprisingly, this solution matches the usual correct solution because so far we have not injected any inconsistency and so, for example, the preferences s_1 and s_2 play no role.

Next, we illustrate three cases of injection of inconsistency into the puzzle. Since complete models tend to be rather large, we show only *tuple*/6 and $\top$ -predicates.

43 The Ukrainian lives in the middle house — Variation 1

house_nationality(3, ukrainian) : **t**.

There are two models:

$m_1 = \{ \text{tuple}(1, \text{yellow}, \text{norwegian}, \text{water}, \text{kools}, \text{zebra}) : \mathbf{t}$
 $\text{tuple}(2, \text{blue}, \text{japanese}, \text{tea}, \text{parliament}, \text{horse}) : \mathbf{t}$
 $\text{tuple}(3, \text{ivory}, \text{ukrainian}, \text{milk}, \text{old_gold}, \text{snails}) : \mathbf{t}$
 $\text{tuple}(4, \text{green}, \text{spaniard}, \text{coffee}, \text{chesterfield}, \text{dog}) : \mathbf{t}$
 $\text{tuple}(5, \text{red}, \text{englishman}, \text{orange_juice}, \text{lucky_strike}, \text{fox}) : \mathbf{t}$
 $\text{house_nationality}(3, \text{ukrainian}) : \top \}$

$m_2 = \{ \text{tuple}(1, \text{yellow}, \text{norwegian}, \text{tea}, \text{kools}, \text{zebra}) : \mathbf{t}$
 $\text{tuple}(2, \text{blue}, \text{japanese}, \text{water}, \text{parliament}, \text{horse}) : \mathbf{t}$
 $\text{tuple}(3, \text{ivory}, \text{ukrainian}, \text{milk}, \text{old_gold}, \text{snails}) : \mathbf{t}$
 $\text{tuple}(4, \text{green}, \text{spaniard}, \text{coffee}, \text{chesterfield}, \text{dog}) : \mathbf{t}$
 $\text{tuple}(5, \text{red}, \text{englishman}, \text{orange_juice}, \text{lucky_strike}, \text{fox}) : \mathbf{t}$
 $\text{house_nationality}(3, \text{ukrainian}) : \top \}$

The puzzle originally implied that the Ukrainian lives in the second house. Therefore, Variation 1 generates an inconsistency about the Ukrainian being in the middle house. By Sentence 20, we cannot derive *house_drink*(3, tea) : **t** because propagation of inconsistency is blocked in this case. This sanctions two possibilities: one where *house_drink*(1, water) : **t** and *house_drink*(2, tea) : **t** hold and the other where *house_drink*(1, tea) : **t** and *house_drink*(2, water) : **t** hold.

44 The Lucky Strike is smoked in the middle house — Variation 2

house_smoke(3, lucky_strike) : **t**.

Again, we have two models:

$m_1 = \{ \text{tuple}(1, \text{yellow}, \text{norwegian}, \text{water}, \text{kools}, \text{fox}) : \mathbf{t}$
 $\text{tuple}(2, \text{blue}, \text{ukrainian}, \text{tea}, \text{chesterfield}, \text{horse}) : \mathbf{t}$
 $\text{tuple}(3, \text{ivory}, \text{spaniard}, \text{milk}, \text{lucky_strike}, \text{dog}) : \mathbf{t}$
 $\text{tuple}(4, \text{green}, \text{japanese}, \text{coffee}, \text{parliament}, \text{zebra}) : \mathbf{t}$
 $\text{tuple}(5, \text{red}, \text{englishman}, \text{orange_juice}, \text{old_gold}, \text{snails}) : \mathbf{t}$
 $\text{house_smoke}(3, \text{lucky_strike}) : \top \}$

$m_2 = \{ \text{tuple}(1, \text{yellow}, \text{norwegian}, \text{orange_juice}, \text{kools}, \text{fox}) : \mathbf{t}$

tuple(2, blue, ukrainian, tea, chesterfield, horse) : t
tuple(3, ivory, spaniard, milk, lucky_strike, dog) : t
tuple(4, green, japanese, coffee, parliament, zebra) : t
tuple(5, red, englishman, water, old_gold, snails) : t
house_smoke(3, lucky_strike) : ⊤

The puzzle originally implied that the Lucky Strike is smoked in the fourth house. As a result, Variation 2 generates an inconsistency regarding the Lucky Strike being smoked in the middle house. By Sentence 38, we cannot derive *house_drink(3, orange_juice) : t* because propagation of inconsistency is blocked by the epistemic implication. This sanctions two possibilities: one where *house_drink(1, water) : t* and *house_drink(5, orange_juice) : t* hold and the other where *house_drink(1, orange_juice) : t* and *house_drink(5, water) : t* hold.

45 Milk is not drunk in the middle house — Variation 3

house_drink(3, milk) : f.

Variation 3 generates the same model as Zebra Puzzle's original solution except for an additional inconsistent fact *house_drink(3, milk) : ⊤*. This is because the puzzle originally implied that milk is drunk in the middle house.

Appendix C Marathon Puzzle in APC_{LP} with Inconsistency Injections

We present here a complete APC_{LP} encoding of Marathon Puzzle (C. Guéret and Sevaux 2000; Schwitter 2012). As with the previous puzzles, the encoding highlights the principles of Section 6 and we also discuss several cases of inconsistency injection.

Marathon puzzle is as follows:

Dominique, Ignace, Naren, Olivier, Philippe, and Pascal have arrived as the first six at the Paris marathon.
 Olivier has not arrived last.
 Dominique, Pascal and Ignace have arrived before Naren and Olivier.
 Dominique who was third last year has improved this year.
 Philippe is among the first four.
 Ignace has arrived neither in second nor third position.
 Pascal has beaten Naren by three positions.
 Neither Ignace nor Dominique are in the fourth position.

The original description implies some implicit background knowledge. First, no runners arrive at the same time (i.e., each runner has a unique arrival position and vice versa). Second, a runner arrives before another runner if the first runner's position number is smaller than the second runner's position.

There is a unique solution for Marathon Puzzle where Ignace arrives first, followed by Dominique, Pascal, Philippe, Olivier, and Naren, in that order. Next, we show the encoding of Marathon puzzle.

- 1 Dominique, Ignace, Naren, Olivier, Philippe, and Pascal have arrived as the first six at the Paris marathon.
runner(dominique) : t. runner(naren) : t. runner(ignace) : t. runner(olivier) : t.
runner(philippe) : t. runner(pascal) : t.

$position(1) : \mathbf{t}. \ position(2) : \mathbf{t}. \ position(3) : \mathbf{t}. \ position(4) : \mathbf{t}. \ position(5) : \mathbf{t}.$
 $position(6) : \mathbf{t}.$

Sentences 2 and 3 are encoded based on Principles 5 and 6. This will cause the second rule in Sentence 3 to be repeated as part of the encoding of Sentence 2 so, as before, we omit the duplicate.

2 Every runner has exactly one position.

$1\{has_position(R, P) : \mathbf{t} \text{ if } position(P) : \mathbf{t}\}1 \leftarrow runner(R) : \mathbf{t}.$

$has_position(R, P) : \mathbf{f} \leftarrow runner(R) : \mathbf{t} \wedge position(P) : \mathbf{t} \wedge \neg has_position(R, P) : \mathbf{t}.$

3 Every position belongs to exactly one runner.

$1\{has_position(R, P) : \mathbf{t} \text{ if } runner(R) : \mathbf{t}\}1 \leftarrow position(P) : \mathbf{t}.$

Encoding of Sentence 4 is straightforward.

4 Olivier has not arrived last.

$has_position(olivier, 6) : \mathbf{f}.$

Sentence 5 is encoded by three rules. The first and second rules are the result of applying Principle 2 defining the *before*-relation based on the position of runners. The third rule represents complete knowledge of the *before*-relation based on Principle 5.

5 If a runner R1 has a position P1 and another runner R2 has a position P2 and $P1 < P2$ then R1 is *before* R2.

$before(R1, R2) : \mathbf{t} \leftarrow runner(R1) : \mathbf{t} \wedge position(P1) : \mathbf{t} \wedge has_position(R1, P1) : \mathbf{t} \wedge$
 $runner(R2) : \mathbf{t} \wedge position(P2) : \mathbf{t} \wedge has_position(R2, P2) : \mathbf{t} \wedge$
 $P1 < P2 \wedge \neg runner(R1) : \top \wedge \neg runner(R2) : \top \wedge$
 $\neg position(P1) : \top \wedge \neg position(P2) : \top \wedge$
 $\neg has_position(R1, P1) : \top \wedge \neg has_position(R2, P2) : \top.$

$before(R2, R1) : \mathbf{f} \leftarrow runner(R1) : \mathbf{t} \wedge position(P1) : \mathbf{t} \wedge has_position(R1, P1) : \mathbf{t} \wedge$
 $runner(R2) : \mathbf{t} \wedge position(P2) : \mathbf{t} \wedge has_position(R2, P2) : \mathbf{t} \wedge$
 $P1 < P2 \wedge \neg runner(R1) : \top \wedge \neg runner(R2) : \top \wedge$
 $\neg position(P1) : \top \wedge \neg position(P2) : \top \wedge$
 $\neg has_position(R1, P1) : \top \wedge \neg has_position(R2, P2) : \top.$

$before(R1, R2) : \mathbf{f} \leftarrow runner(R1) : \mathbf{t} \wedge runner(R2) : \mathbf{t} \wedge \neg before(R1, R2) : \mathbf{t}.$

Encoding of Sentences 6 - 9 is straightforward.

6 Dominique, Pascal and Ignace have arrived before Naren and Olivier.

$before(dominique, naren) : \mathbf{t}. \ before(pascal, naren) : \mathbf{t}. \ before(ignace, naren) : \mathbf{t}.$

$before(dominique, olivier) : \mathbf{t}. \ before(pascal, olivier) : \mathbf{t}. \ before(ignace, olivier) : \mathbf{t}.$

7 Dominique who was third last year has improved this year.

$has_position(dominique, 1) : \mathbf{t} \vee has_position(dominique, 2) : \mathbf{t}.$

8 Philippe is among the first four.

$has_position(philippe, 1) : \mathbf{t} \vee has_position(philippe, 2) : \mathbf{t} \vee$

$has_position(philippe, 3) : \mathbf{t} \vee has_position(philippe, 4) : \mathbf{t}.$

9 Ignace has arrived neither in second nor third position.

$has_position(ignace, 2) : \mathbf{f}.$

$has_position(ignace, 3) : \mathbf{f}.$

Encoding of Sentence 10 is based on Principle 2. The first rule encodes that if the position of Pascal is known, then the position of Naren is the position of Pascal plus 3. The second rule encodes that if the position of Naren is known, then the position of Pascal is the position of Naren minus 3.

10 Pascal has beaten Naren by three positions.

$$\begin{aligned} \text{has_position}(\text{naren}, P2) : \mathbf{t} &\leftarrow \text{has_position}(\text{pascal}, P1) : \mathbf{t} \wedge \text{position}(P1) : \mathbf{t} \wedge \\ &\text{position}(P2) : \mathbf{t} \wedge P2 = P1 + 3 \wedge \neg \text{position}(P1) : \top \wedge \\ &\neg \text{position}(P2) : \top \wedge \neg \text{has_position}(\text{pascal}, P1) : \top. \\ \text{has_position}(\text{pascal}, P1) : \mathbf{t} &\leftarrow \text{has_position}(\text{naren}, P2) : \mathbf{t} \wedge \text{position}(P1) : \mathbf{t} \wedge \\ &\text{position}(P2) : \mathbf{t} \wedge P2 = P1 + 3 \wedge \neg \text{position}(P1) : \top \wedge \\ &\neg \text{position}(P2) : \top \wedge \neg \text{has_position}(\text{naren}, P2) : \top. \end{aligned}$$

Encoding of Sentence 11 is straightforward.

11 Neither Ignace nor Dominique are in the fourth position.

$$\text{has_position}(\text{ignace}, 4) : \mathbf{f}. \quad \text{has_position}(\text{dominique}, 4) : \mathbf{f}.$$

Next we define the consistency preference relation $<_{\mathcal{S}}$, where $\mathcal{S} = (s_1, s_2, \mathcal{B}_{\top})$, which implements Principle 4. Here s_1 says that we hold greater confidence in the information about the *runner*- and *position*-facts and therefore these types of facts are less likely to be inconsistent. The preference s_2 says that next we are likely to be confident in the information given by the *has_position*- and *before*-facts. The last component in $\mathcal{S}$, $\mathcal{B}_{\top}$, is the usual default that gives preference to the most e-consistent models.

$$\begin{aligned} s_1 &= \{\text{runner}(\text{dominique}) : \top, \text{runner}(\text{naren}) : \top, \text{runner}(\text{ignace}) : \top, \\ &\quad \text{runner}(\text{olivier}) : \top, \text{runner}(\text{philippe}) : \top, \text{runner}(\text{pascal}) : \top, \\ &\quad \text{position}(1) : \top, \dots, \text{position}(6) : \top\}. \\ s_2 &= \{\text{has_position}(\text{dominique}, 1) : \top, \dots, \text{has_position}(\text{dominique}, 6) : \top, \dots \\ &\quad \text{has_position}(\text{pascal}, 1) : \top, \dots, \text{has_position}(\text{pascal}, 6) : \top, \\ &\quad \text{before}(\text{dominique}, \text{naren}) : \top, \dots, \text{before}(\text{dominique}, \text{pascal}) : \top, \dots \\ &\quad \text{before}(\text{pascal}, \text{dominique}) : \top, \dots, \text{before}(\text{pascal}, \text{philippe}) : \top\}. \end{aligned}$$

There is a single most consistency-preferred model as expected. The puzzle, as stated, has no inconsistent information so we show only the *has_position*-facts.

$$m = \{\text{has_position}(\text{ignace}, 1) : \mathbf{t}, \text{has_position}(\text{dominique}, 2) : \mathbf{t}, \\ \text{has_position}(\text{pascal}, 3) : \mathbf{t}, \text{has_position}(\text{philippe}, 4) : \mathbf{t}, \\ \text{has_position}(\text{olivier}, 5) : \mathbf{t}, \text{has_position}(\text{naren}, 6) : \mathbf{t}\}.$$

In addition, there are many *before*-facts which encode the sequential order of arrival of the runners.

Next, we illustrate several cases of injection of inconsistency into the puzzle. Since complete models tend to be rather large, we show only the *has_position*- and $\top$ -predicates.

12 Pascal arrives in the sixth position — Variation 1

$$\text{has_position}(\text{pascal}, 6) : \mathbf{t}.$$

There is one model and it contains inconsistencies:

$$m = \{\text{has_position}(\text{ignace}, 1) : \mathbf{t}, \text{has_position}(\text{dominique}, 2) : \mathbf{t}, \\ \text{has_position}(\text{naren}, 3) : \mathbf{t}, \text{has_position}(\text{philippe}, 4) : \mathbf{t}, \\ \text{has_position}(\text{olivier}, 5) : \mathbf{t}, \text{has_position}(\text{pascal}, 6) : \mathbf{t},$$

$before(pascal, naren) : \top, before(pascal, olivier) : \top$.

The *before*-facts still represent the sequential order of arrival, so $before(pascal, naren) : \mathbf{t}$ and $before(pascal, olivier) : \mathbf{t}$ are true in the model. However, Sentence 12 contradicts these facts, so $before(pascal, naren) : \mathbf{f}$ and $before(pascal, olivier) : \mathbf{f}$ are also true. Therefore, these facts become inconsistent. Sentence 10 does not imply any inconsistencies beyond the ones already mentioned.

13 Ignace arrives in the second position. — Variation 2

$has_position(ignace, 2) : \mathbf{t}$.

There is one model:

$m = \{has_position(dominique, 1) : \mathbf{t}, has_position(ignace, 2) : \top,$
 $has_position(pascal, 3) : \mathbf{t}, has_position(philippe, 4) : \mathbf{t},$
 $has_position(olivier, 5) : \mathbf{t}, has_position(naren, 6) : \mathbf{t}\}.$

Given that Sentence 13 contradicts Sentence 9, the fact $has_position(ignace, 2)$ becomes inconsistent and the order of arrival of Dominique and Ignace are swapped.

14 Philippe arrives before Dominique. — Variation 3

$before(philippe, dominique) : \mathbf{t}$.

Now, we get 4 models:

$m_1 = \{has_position(ignace, 1) : \mathbf{t}, has_position(dominique, 2) : \mathbf{t},$
 $has_position(pascal, 3) : \mathbf{t}, has_position(philippe, 4) : \mathbf{t},$
 $has_position(olivier, 5) : \mathbf{t}, has_position(naren, 6) : \mathbf{t},$
 $before(philippe, dominique) : \top\}.$

$m_2 = \{has_position(philippe, 1) : \mathbf{t}, has_position(dominique, 2) : \mathbf{t},$
 $has_position(pascal, 3) : \mathbf{t}, has_position(ignace, 4) : \top,$
 $has_position(olivier, 5) : \mathbf{t}, has_position(naren, 6) : \mathbf{t}\}.$

$m_3 = \{has_position(philippe, 1) : \mathbf{t}, has_position(dominique, 2) : \mathbf{t},$
 $has_position(pascal, 3) : \mathbf{t}, has_position(olivier, 4) : \mathbf{t},$
 $has_position(ignace, 5) : \mathbf{t}, has_position(naren, 6) : \mathbf{t},$
 $before(ignace, olivier) : \top\}.$

$m_4 = \{has_position(philippe, 1) : \mathbf{t}, has_position(dominique, 2) : \mathbf{t},$
 $has_position(naren, 3) : \mathbf{t}, has_position(pascal, 4) : \mathbf{t},$
 $has_position(ignace, 5) : \mathbf{t}, has_position(olivier, 6) : \top,$
 $before(pascal, naren) : \top, before(ignace, naren) : \top\}.$

The first model generates the same solution as the original puzzle except for the inconsistency where Philippe arrives before Dominique. The second model places Ignace in the fourth position, which contradicts Sentence 11, so placing Ignace in the fourth position becomes inconsistent. The third model places Olivier before Ignace, which is in contradiction with Sentence 6. All the other constraints are satisfied, so no more inconsistencies are derived. The fourth model places Naren before Pascal and Ignace, which contradicts Sentence 6. Besides, it places Olivier the last, which contradicts Sentence 4.