Reasoning About Applicable Law in Private International Law in Logic Programming

Ken SATOH a,2 and Matteo BALDONI b Laura GIORDANO c
a National Institute of Informatics, Japan b Università di Torino, Italy c Università del Piemonte Orientale, Italy

Abstract. We formalized renvoi in private international law in JURIX 2019 in terms of modal logic fragment. In this demonstration paper, we show an implementation of the formalism by translating modal formula into a logic program.

1. Introduction

Private international law (PIL) “enables the coexistence of multiple normative systems, having distinct and often contradictory rules [3]”. In international legal relations, we have to decide “applicable law” for an international legal issue since the issue is related with multiple countries which might have contradictory rules each other for the issue. We formalize PIL to decide “applicable law” for an international legal matter in one country may require to refer to another law in another country which may result in a sequence of references of laws to different countries. We formalize this reasoning by a rule-based fragment of the modal language in [1], extended with context variables, and allows the interactions among contexts to be captured, context variables to occur within modalities and context names to be used as predicate arguments, thus supporting a simple combination of meta-predicates and modal constructs. In this paper, we translate the formalism into a logic program which refines predicate to express legal matters with a variable to express the country of applicable law for the international legal matters. As a related research, the specificity of the rules in Conflict of Laws have been analyzed by Markovich [5] in the formalism of the input/output framework [4], where such rules assign a set of sets of norms (a legal system) to a given domain (a set of statements).

2. Reasoning about Applicable Law

We review how to reason about applicable law defined in [2]. Given a legal matter P in one country, C, we would like to decide whether the matter is valid in the country in the following way.
1. We decide the country $X$ whose law is applied to decide the matter $P$ as follows.
   (a) There should be a rule in the private international law in $C$ which indicates an applicable law in (possible another) country $C'$ for the matter $P$ in the country, $C$.
   (b) If $C' = C$, $X = C$.
   (c) Else ($C' \neq C$), we need to again decide the country $X$ of the applicable law for $P$ according to the private international law in $C'$ (called “renvoi”).
   (d) If we detect a loop in the “renvoi”, we set the applicable law to the starting country of the loop. For example, if the private international laws makes this reference of applicable law, "A $\rightarrow$ B $\rightarrow$ C $\rightarrow$ D $\rightarrow$ B'", then we decide an applicable law for the matter as country B.

2. We decompose the matter $P$ into submatters according to a rule defined in the applicable law in $X$.

3. If a submatter is determined by a global fact and the global fact is in the fact base, the submatter is valid.

4. Otherwise, we iterate the process above (we decide an applicable law of the submatter and then check the submatter is valid in the applicable law).

3. Translation of Modal Program into a Logic Program

Top rule is translated into:

```
holds(P\#CountryA) :- applicable_law(P,CountryB,[])\#CountryA, P\#CountryB.
```

which means that

1. we reason about renvoi, that is, decide a country for an applicable law, CountryB whose law determines a matter, $P$ in CountryA, then
2. we determine $P$ in the law of CountryB.

Here is a rule to compute applicable law to handle a loop case in renvoi:

```
applicable_law(P,CountryA,ReferredHistory)#CountryA :-
    member(CountryA,ReferredHistory).
```

means that if the current country is in the sequence, ReferredHistory of referred country so far, then the country of applicable law is CountryA.

Here is a rule to computing applicable law for specific predicate:

```
applicable_law(P,CountryC,ReferredHistory)#CountryA :-
    \+member(CountryA,ReferredHistory),
    some_conditions_to_refer_to_other_country(CountryB),
    applicable_law(P,CountryC,[CountryA\|ReferredHistory])#CountryB.
```

For example, to decide a country for an applicable law for heir(Child,Parent), a condition to refer to other country is home_country(Parent,CountryB) so a rule of computing an applicable law is:

```
applicable_law(heir(Child,Parent),CountryC,ReferredHistory)#CountryA :-
    \+member(CountryA,ReferredHistory),
    home_country(Parent,CountryB),
    applicable_law(
        heir(Child,Parent),CountryC,[CountryA\|ReferredHistory])#CountryB.
```
Global rules and facts represented in the modal settings, “□{(H := B1, . . . , Bn).}” or “□H.” is translated into ordinary horn clauses: “H := B1, . . . , Bn.” or “H.” Domestic rules specific to a country, Country, “□[Country] {H := B1, . . . , Bn.}” is translated into H#Country := B1', . . . , Bn'. where Bi' is Bi#Country if Bi is the head of a domestic rule, otherwise (Bi is the head of a global rule), Bi.

Fig. 1 shows the entire program to compute heir(taro,john)#japan. In the program, holds(P#Country) and P#Country are similar predicates but P#Country means that a predicate, P, is true according to the law in a country, Country whereas holds(P#Country) means that a predicate P is true according to a law in a country, Country’ after we find that the an applicable law of the Country’ for P starting from Country.

4. Demonstration

We will demonstrate a process to reason about applicable law for the example in [2]. Fig. 2 is an output of how to reason about heir(taro,john) in Japan. In this example, firstly we decide a country whose law is applied (we call applied country here) to this issue. We assume that a rule for this decision is universal in that the applied country is the home country of a parent derived from the facts of parent’s nationality and habitual residence. In this example, it is country1. Then, we check heir(taro,john) is true in country1. Then according to the law of country1, we should check legitimate_child_parent_rel(Child,Parent) in country1. Then, again we decide an applied country for legitimate child-parent relation and we find that country1 should be the applied country. Then, legitimate child-parent relation is true if the parent is married and biological relation between the parent and the child exits.
To show the marriage status, we need to find an applied country for the marriage status and we find that the applied country is country1. According to a universal rule, we can show the marriage status by showing agreement between a pair and registration.

5. Conclusion

We show a translation method of modal formalization of reasoning about applicable law into logic programming.
References

[1] Baldoni M, Giordano L, Martelli A. A modal extension of logic programming: modularity, beliefs and hypothetical reasoning. J. Log. Comput., Vol. 8, No. 5. 1998. p 597-635.
[2] Baldoni M, Giordano L, Satoh, K. Renvoi in private international law: a formalization with modal context. Proc. of JURIX 2019. 2019. p157-162.
[3] Dung PM, Sartor G. The modular logic of private international law. Artif. Intell. Law, Vol. 19, No. 2-3. 2011. p233-261.
[4] Makinson D, van der Torre L. What is input/output logic? Foundations of the Formal Sciences II. 2003. p163-174.
[5] Markovich R. On the formal structure of rules in conflict of laws. Proc. of JURIX 2019. 2019. p199-204.