Role Lists

by Sven Nilsen, 2020

In this paper I introduce the idea of a role list, which is a compact way of representing roles in the theory of unique universal binary relations. Role lists can be used to relax the universal property.

Assume that one has a system modeled with unique universal binary relations.
A role list is a list of roles that consists of all solutions of the following problem:

```
why(∃ x { role(x, role) })
```

Alternative syntax is `role_of` instead of `role`

Such that the list contains all and only those of this constraint:

```
∀ x { role(x, role) }
```

One can use modal logic to describe this formally:

```
∀ x { □role(x, role)  }
```

This list must include `role`, because the role of roles is role, but otherwise one is free to choose.

For example:

```
role
type
value
codomain
domain
```

One application of role lists, is to relax the constraints of unique universal binary relations a bit, such that if `(a, b)` collides with the role of `(c, b)`, then the collision can be resolved without problems.

To explain how this works, one can use the intuition from Avatar Graphs.

For any role list, one can construct “avatars” for objects automatically. These objects behave exactly like the “core self”, except that they play a role for incoming relations. Since the role list is avaible, it is possible to enumerate all roles and therefore also enumerate all avatars of a core self. If an object has more than one avatar, then it is implicitly understood that the universal property is relaxed.

This relationship dynamics keeps the semantic complexity down, without needing to explicitly formalize this in detail. For any finite model, an algorithm to compute the semantic complexity is decidable. However, most of the time, one is not interested in the semantic complexity, but other properties of the model. Therefore, it is convenient to know that a lower upper estimate of the semantic complexity is available, given enough work, than instead of relying on getting all the details right in a much higher semantic complexity.