Project title

Demonstration tool for teaching and learning of bisimulation algorithm for semistructured databases

Supervisor

Vladimir Sazonov, Logic and Computations Group,
http://www.csc.liv.ac.uk/˜sazonov/

Brief description

The goal of this Project is to implement *bisimulation algorithm for semistructured databases* [1, 3, 2, 4] in such a way that to make it a demonstration tool for students studying this topic. Semistructured (or Web-like) databases are represented in terms of graphs with labelled edges (called also *labelled transition systems* with labels currying the database information). In this approach graphs are considered up to bisimulation relation on the nodes where bisimulation intuitively means also "deep" *informational equivalence* of WDB data.

More detailed description (cf. op. cit., especially [4] and also Slides of COMP311)

A Web-like or semistructured database under hyperset approach is a generalisation of the ordinary approach to relational databases. Any data under this approach is considered as a finite set of labelled elements

$$\{label_1 : x_1, \dots, label_n : x_n\}, \text{ for example,}$$
$$\{\texttt{Student} : s, \texttt{Department} : d, \texttt{StartAndEndOfStudy} : t\}.$$

The order and repetition of the labelled elements $label_i : x_i$ does not matter. Labels also correspond to *attributes* of relational databases (actually, strings of symbols — not sets). Labelled elements $x_1, \dots, x_n$ or s, d, t are themselves complex data (sets). Say,

$$s = \{\texttt{Name} : n, \texttt{Birthdate} : b, \texttt{Address} : a\}.$$

For uniformity, *atomic data* are also represented as a labelled *singleton set* $\{\texttt{label} : \emptyset\}$ where $\emptyset = \{\}$, or abbreviated as "label". Then, we can have, for example,

$$n = \{\texttt{FirstName} : \texttt{"David"}, \texttt{LastName} : \texttt{"Beckham"}\}.$$

Admitting an analogy with WWW, a set is an analogue of a *Web page*, labelled elements of a set are *hyperlinks* (labels can be 'clicked' to 'download' corresponding Web pages—sets). Then, as in WWW, cyclic (hyper) sets like $\Omega = \{\Omega\}$ are allowed. We assume here an 'empty' label $\Box$ before the element Ω, which is usually omitted. *Pure* hypersets are those which contain no labels at any depth (or, equivalently, only the empty label $\Box$).

Any *datbase state* (i.e., a hyperset) is represented in a computer as a system of equations, as above or as a directed edge labelled graph Thus, s considered as a vertex of the graph has three outgoing labelled edges: $s \xrightarrow{\texttt{Name}} n$, $s \xrightarrow{\texttt{BirthDate}} b$, and $s \xrightarrow{\texttt{Address}} a$. Vertics also correspond to URLs of a Web page, and the labelled sets, such as $\{label_1 : x_1, \dots, label_n : x_n\}$, correspond to HTML files. In a browser we see only (clickable) labels $\underline{label_1}, \dots, \underline{label_n}$ with corresponding URLs $x_1, \dots, x_n$ *hidden*.

Bisimulation relation. Now, assume we have another strudent s' described by equations

$$s' = \{\texttt{Birthdate} : b', \texttt{Name} : n', \texttt{Address} : a'\}.$$

$$n' = \{\texttt{LastName} : \texttt{"Beckham"}, \texttt{FirstName} : \texttt{"David"}, \texttt{LastName} : \texttt{"Beckham"}\}.$$

From set-theoretic point of view $n = n'$ despite repetition of elements in n' and the order. Assume also $a = a'$ and $b = b'$. Then we can intuitively conclude that $s = s'$. We also say that these sets are bisimular, writing this alternatively as $s \approx s'$. The example considered is quite trivial. More complicated example presented at the end of this text, and the precise definition/algorithm for bisimulation relation is presented e.g. in the slides for COMP311 available from the home page of Vladimir Sazonov.

To implement bisimulation and to demonstrate how the implementation works a student should represent WDB in terms of systems of equations of the above kind (as one text file). The bisimulation algorithms should work with all set names of a given system of equations and demonstrate to a user all steps of derivation of non-bisimilar set names. At the end, the information on which set names are bisimilar should be used for simplifying the initial system of set equations. All of this should be done and visualised in such a way that any newcomer would understand how the algorithm works.

As the result, this should be a good demonstrating tool for teaching and learning with a number of instructive examples of systems of set equations which, however, should work for arbitrary such systems.

For a student working on this Project this would be a good opportunity to apply his/her programming skills and ambitions, and to learn new ideas on unstructured databases — a fresh and promising direction of research.

Background requirements

Familiarity with the simplest concepts of logic, simplest set-theoretic concepts, very general ideas on databases, having sufficiently advanced programming skills and, possibly, theoretical inclinations. It is very desirable to read some parts of slides to Module COMP311 especially devoted to semistructured or Web-like databases.

References

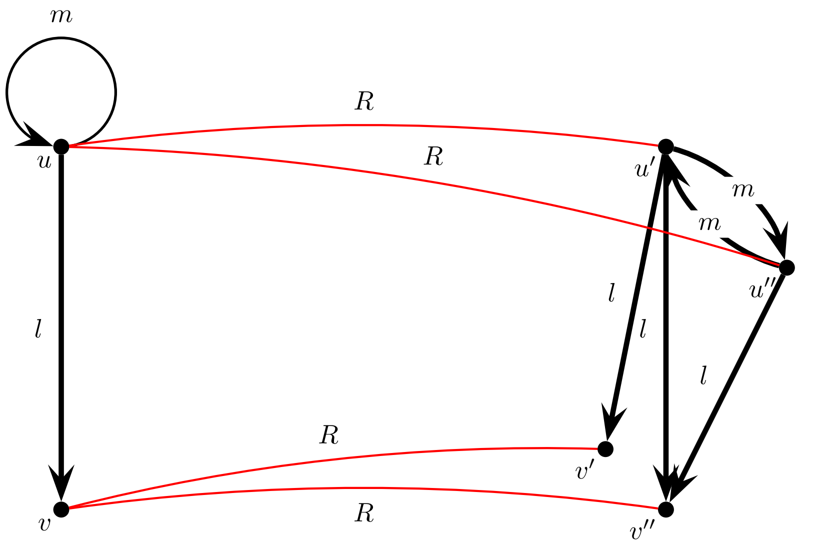
[1] S. Abiteboul, P. Buneman and D. Suciu, *Data on the Web; From Relations to Semistructured Data and XML*, Morgan Kaufmann Publishers, 2000.

[2] Lisitsa, A., and Sazonov, V., Bounded Hyper-set Theory and Web-like Data Bases. *Computational Logic and Proof Theory, 5th Kurt Gödel Colloquium, KGC'97*, Springer LNCS Vol. 1289, 1997, pp. 172–185.

[3] Sazonov, V.Yu.: 1993, 'Hereditarily-finite sets, data bases and polynomial-time computability', *Theoretical Computer Science* **Vol. 119**, Elsevier, pp. 187–214.

[4] Sazonov, V.Yu., Querying Hyperset/Web-like Databases, Logic Journal IGPL, 2006; 14(5): 785-814.

Example of a bisimulation relation R between two graphs



m
R
R
u
u'
m
m
u''
l
l
l
l
R
v'
v
R
v''