Web Sites Verification: an Abductive Logic Programming Tool

P. Mancarella[1], G. Terreni[1], and F. Toni[2]

[1] Dipartimento di Informatica, Università di Pisa, Italy
[2] Department of Computing, Imperial College London, UK
Email: {paolo,terreni}@di.unipi.it; ft@doc.ic.ac.uk

Abstract. We present the CIFFWEB system, an innovative tool for the verification of web sites, relying upon abductive logic programming. The system allows the user to define rules that a web site should fulfill by using (a fragment of) the query language Xcerpt. The rules are translated into abductive logic programs with constraints and their fulfillment is checked through the CIFF abductive proof procedure.

1 Web Checking Rules

The exponential growth of the WWW raises the question of maintaining and repairing automatically web pages at both structural and data level. Our *web checking rules* are characterized by using (a fragment of) the query language Xcerpt [2] for expressing complex queries in a natural syntax. The following is an example of a XML page **[P1]** about a theater company (left) and a rule expressing that *no director should occur twice in the director list* **[R1]** (right).

```
%%% directorindex.xml
<dirlist>
 <dir>dir1</dir>
 <dir>dir2</dir>
 <dir>dir2</dir>
</dirlist>
```

```
GOAL all error [var D,"double director"]
FROM
in {resource{"file:directorindex.xml"},
    dirlist {{
    dir {{var D}}, dir {{var D}} }}
} END
```

Web checking rules are specified by a *condition part* (starting with `FROM`) and an *error part* (starting with `GOAL`). The intuitive meaning of a rule is that for each instance in the XML `resource`(s) matching the condition part, an error needs to be returned. Due to lack of space, the example does not cover the whole expressiveness of our rules: in particular the absence of XML data (`without` construct) and (arithmetical) constraints over variables can also be expressed.
However, to the best of our knowledge, Xcerpt lacks of both a clear semantics for negation constructs (`without`) and a concrete tool for evaluating Xcerpt queries. Hence, we map web checking rules into *abductive logic programs with constraints* (ALPCs) that can be fed as input to the CIFF System 4.0, an implementation of the general-purpose CIFF abductive proof procedure [3] which is sound with respect to the 3-valued completion semantics. Using CIFF for determining the fulfillment of the rules, we inherit its formal properties, thus obtaining a sound concrete tool for web sites verification.

2 Mapping rules to abductive logic programs

The CIFF proof procedure gets as input an ALPC $\langle P, A, IC\rangle_{\Re}$ where P is a normal logic program (with constraints), A is a set of *abducible predicates* and IC is a set of *integrity constraints* of the form $L_1 \wedge \cdots \wedge L_m \rightarrow H$ where each L_i is a literal and H is a disjunction of atoms. Constraint atoms are evaluated wrt an underlying structure $\Re$, as in constraint logic programming [3].
Since CIFF is not designed to handle directly XML resources, we translate XML pages into sets of atoms of the form `pg_el(ID,Tag,IDF)`. Each XML `Tag` element is associated to both a unique `ID` and to its father's id (`IDF`) in order to represent the page structure. The same is done for the data inside a XML tag. Similarly, we translate Xcerpt rules into ALPCs, where abducibles are associated with *errors*. In the absence of negation (`without` construct) each web checking rule is translated into a single integrity constraint whose head is an abducible atom `abd_err(Args,Msg)` representing the error. As an example, the translation of both **[P1]** and **[R1]** is the following.

```
[P1]: pg_el(1,dirlist,_). pg_el(2,dir,1). data_el(3,'dir1',2).
      pg_el(4,dir,1). data_el(5,'dir2',4).
      pg_el(6,dir,1). data_el(7,'dir2',6).

[R1]: [pg_el(ID1,dirlist,_), pg_el(ID2,dir,ID1), data_el(ID3,D,ID2),
       pg_el(ID4,dir,ID1), data_el(ID5,D,ID4), ID2 #\= ID4]
      implies   [abd_err([D],"double director")].
```

Running the CIFF System 4.0 with the above input, the abductive answer `[abd_err(['dir2'],'double director')]` is correctly produced[3].

3 Conclusions

The CIFFWEB tool shows how abductive logic programming can be exploited for verifying properties' fulfillment of web sites. The main advantages of this approach are the expressiveness, a clear formal semantics and a concrete computational counterpart. There are many issues to be addressed yet. In particular abductive reasoning can also be exploited for web sites repairing. A (more complex) formalization of the repairing task is work in progress. We are also working on expressiveness extensions to the framework, a user-friendly GUI, and medium-large size experiments. In the literature there is limited work on these topics. The GVERDI-R system [1] is the closest work.

References

1. M. Alpuente, D. Ballis, and M. Falaschi. A rewriting-based framework for web sites verification. *Electronic Notes in Theoretical Computer Science*, 124:41–61, 2005.
2. F. Bry and S. Schaffert. The XML query language Xcerpt: Design principles, examples, and semantics, 2002.
3. U. Endriss, P. Mancarella, F. Sadri, G. Terreni, and F. Toni. The CIFF proof procedure for abductive logic programming with constraints. In *Proc. JELIA*, 2004.

[3] The CIFF System 4.0 and the full version of this paper with all the technical details are both available at `www.di.unipi.it/~terreni/research.php`