

Rewriting-based navigation of Web sites¹

Salvador Lucas^a

^a *DSIC, UPV, Camino de Vera s/n, 46022 Valencia, Spain. slucas@dsic.upv.es*

Abstract

In this paper we sketch the use of term rewriting techniques for modeling the dynamic behavior of Web sites.

Key words: Hypertext browsing, Semantic modeling of Web sites, Term rewriting.

1 Introduction

The World Wide Web (WWW) provides easy and flexible access to information and resources distributed all around the world. Although Web sites are usually connected via Internet, many hypertext-based systems like on-line help in compilers, programming language reference manuals, electronic books, or software systems are now organized in a very similar way, also using the same description language (HTML) of Web sites. *Browsing* such systems is an essential aspect of their design and use. Having appropriate dynamic models of Web sites is essential for guaranteeing the expected behavioral properties.

Rewriting techniques [BN98,Ohl02,Ter03] have been recently used to reason about the *static* contents of Web sites [ABF05]. In this paper we show that term rewriting techniques are also well-suited for modeling the dynamic behavior of Web sites. We use Maude [CDEL⁺02] as a suitable specification language for the rewriting models which also permits to explore interesting properties like the reachability of Web pages within the site.

2 From ARSs to TRSs

We use a (finite) set of symbols (an alphabet) \mathcal{P} to give name to the Web *pages* of a Web site. Regarding its dynamic modeling, the most relevant information contained in a Web page is, of course, that of the links which

¹ Work partially supported by Spanish MEC grant SELF TIN 2004-07943-C04-02, Acción Integrada HU 2003-0003, and EU-India Cross-Cultural Dissemination project ALA/95/23/2003/077-054.

can originate that a new Web page is downloaded and then used to further browsing the site. The obvious way to express the different transitions between Web pages is to give the (finite) set of transitions among them, i.e., for each Web page p , we can define $\rightarrow_p = \{(p, p_1), \dots, (p, p_{n_p})\} \subseteq \mathcal{P} \times \mathcal{P}$ which is the abstract relation between the page p and its immediate successors (i.e., the pages $p_1, \dots, p_{n_p} \in \mathcal{P}$ which are reachable from p in a single step). The pair $(\mathcal{P}, \rightarrow_{\mathcal{P}})$, where $\rightarrow_{\mathcal{P}} = \bigcup_{p \in \mathcal{P}} \rightarrow_p$ is an *Abstract Reduction System* (ARS [BN98, Chapter 2]) and we can use the associated computational relations $\rightarrow_{\mathcal{P}}, \rightarrow_{\mathcal{P}}^+, \dots$, to describe the dynamic behavior of our Web site. For instance, reachability of a Web page p' from another page p can be rephrased as $p \rightarrow_{\mathcal{P}}^* p'$.

This abstract model is intuitively clear and can, then, be used as a reference for building more elaborated ones. For many applications, however, this ARS-based framework becomes too restrictive. For instance, modeling *safe* (user-sensitive) access to a Web page requires to represent information about the users and modeling some kind of *validation* before granting any access.

Term Rewriting Systems (TRSs [BN98, Ter03]) provide a more expressive setting by allowing the use of *signatures*, i.e., sets of symbols which can be used to build structured objects (terms) by joining terms below a symbol of the signature. For instance, a safe Web page p can take now an argument representing the *user* who is trying to get access to this page. Web pages p containing no link are just constant symbols p (without any transition). Web pages p without safety requirements are represented by rewrite rules $p(U) \rightarrow p_i(U)$ for $1 \leq i \leq n_p$. The definition of a safe page p is as follows:

$$\begin{array}{lll} p(U) \rightarrow vp(U) & vp(u_1) \rightarrow bp(u_1) & bp(U) \rightarrow p_1(U) \\ & \vdots & \vdots \\ & vp(u_{m_p}) \rightarrow bp(u_{m_p}) & bp(U) \rightarrow p_{n_p}(U) \end{array}$$

where vp and bp stand for *validate* and *browse* page p , respectively, and u_i for $1 \leq i \leq m_p$ are terms (e.g., constant symbols) representing the users who are allowed to gain access to the Web page p . The resulting TRS is *shallow* and *linear*²; thus, reachability is decidable [Com00]. Then, reachability of a Web page from another one is decidable too.

Now, after representing the Web site as a Maude rewriting module, it is possible to ask Maude about reachability issues. For instance, the following Maude module provides a partial representation of the WWV'05 site (see <http://www.dsic.upv.es/workshops/wwv05>):

```
mod WebWWV05 is
  sort S .
  ops wwv05 submission speakers org valencia accomodation travelling
  : S -> S .
  ops sbmlink entcs entcswwv05 : S -> S .
```

² A TRS is shallow if variables occur (at most) at depth 1 both in the left- and right-hand sides of the rules [Com00, Section 4]. A TRS is linear if variables occur at most once both in left- and right-hand sides of the rules [BN98, Definition 6.3.1].

```

ops login vlogin blogin : S -> S .
ops forgotten register submit : S -> S .
ops krishnamurthi finkelstein : S -> S .
ops alpuente ballis escobar : S -> S .
op cfp : -> S .
ops slucas smith : -> S .
vars P PS X U : S .
rl wwv05(U) => submission(U) .      rl wwv05(U) => speakers(U) .
rl wwv05(U) => org(U) .              rl wwv05(U) => cfp .
rl wwv05(U) => valencia(U) .        rl wwv05(U) => accomodation(U) .
rl wwv05(U) => travelling(U) .      rl submission(U) => sbmlink(U) .
rl submission(U) => entcs(U) .      rl submission(U) => entcswwv05(U) .
rl sbmlink(U) => login(U) .         rl sbmlink(U) => forgotten(U) .
rl sbmlink(U) => register(U) .      rl speakers(U) => finkelstein(U) .
rl speakers(U) => krishnamurthi(U) . rl org(U) => alpuente(U) .
rl org(U) => ballis(U) .            rl org(U) => escobar(U) .
rl login(U) => vlogin(U) .          rl vlogin(slucas) => blogin(slucas) .
rl blogin(U) => submit(U) .
endm

```

The only safe page is `login`, which grants access to the submission system. For the sake of simplicity, we have omitted many links. In fact, the only ‘terminal’ page is `cfp`, containing the textual version of the WWV’05 call for papers. We can check whether `slucas` (who has been previously registered) can get access to the submission system (page `submit`).

```

Maude> search wwv05(slucas) =>+ submit(slucas) .
search in WebWWV05safe : wwv05(slucas) =>+ submit(slucas) .

```

```

Solution 1 (state 21)
states: 22  rewrites: 21 in 0ms cpu (0ms real) (~ rewrites/second)
empty substitution

```

```

No more solutions.
states: 22  rewrites: 21 in 0ms cpu (1ms real) (~ rewrites/second)

```

Maude tells us that there is only one way for `slucas` to reach the submission page. The command `show path 21` provides the concrete path:

```

wwv05(slucas) → submission(slucas) → sbmlink(slucas)
  → login(slucas) → vlogin(slucas) → blogin(slucas)
  → submit(slucas)

```

The non-registered user `smith` cannot reach this protected part of the site:

```

Maude> search wwv05(smith) =>+ submit(smith) .
search in WebWWV05safe : wwv05(smith) =>+ submit(smith) .

```

```

No solution.
states: 20  rewrites: 19 in 0ms cpu (0ms real) (~ rewrites/second)

```

3 Further improvements and applications

The basic model in Section 2 can be improved in a number of different ways to obtain more expressive models and/or analyze other behavioral issues:

- (i) *Structured* names of users and Web pages allowing for more intuitive and hierarchical naming systems.
- (ii) Efficiency of browsing paths; e.g., shortest path (if any) leading from a Web page to another one.
- (iii) Finiteness of the search space. Of course, the number of pages in a Web site is always finite, but this could eventually be missed in more expressive models. The use of type information and/or syntactic replacement restrictions [Luc02] could be helpful to avoid this problem.
- (iv) Frequency of use of the different links (by applying the recently introduced *probabilistic* approaches to term rewriting).

Finally, the rewriting theory could also benefit from the new research directions pointed by the analysis of the Web. Some challenging aspects are:

- (i) Structured definition of Web sites: a given site can often be considered as composed by many smaller sites. This kind of issues correspond to the analysis of *modular* properties in Term Rewriting [Ohl02], but the current developments are probably too weak for modeling Web site structures.
- (ii) Evolving Web sites: adding new pages to a Web site is quite usual. This corresponds to dynamically adding new rules to the model of the site.

References

- [ABF05] 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.
- [BN98] F. Baader and T. Nipkow. *Term Rewriting and All That*. Cambridge University Press, 1998.
- [CDEL⁺02] M. Clavel, F. Durán, S. Eker, P. Lincoln, N. Martí-Oliet, J. Meseguer, and J. Quesada. Maude: specification and programming in rewriting logic. *Theoretical Computer Science* 285(2):187-243, 2002.
- [Com00] H. Comon. Sequentiality, Monadic Second-Order Logic and Tree Automata. *Information and Computation*, 157(1-2): 25-51, 2000.
- [Luc02] S. Lucas. Context-sensitive rewriting strategies. *Information and Computation*, 178(1):293-343, 2002.
- [Ohl02] E. Ohlebusch. *Advanced Topics in Term Rewriting*. Springer-Verlag, Berlin, 2002.
- [Ter03] TeReSe, editor, *Term Rewriting Systems*, Cambridge Univ. Press, 2003.