Rewriting-Based Navigation of Web Sites: Looking for Models and Logics

Salvador Lucas

DSIC, UPV, Camino de Vera s/n, 46022 Valencia, Spain.

Abstract

In this paper we outline the use of term rewriting techniques for modeling the dynamic behavior of Web sites. We associate rewrite rules to each Web page expressing the Web pages which are immediately reachable from this page. The obtained system permits the application of well-known results from the rewriting theory to analyse interesting properties of the Web site. In particular, we briefly discuss the use of some logics with strong connections with term rewriting as a basis for specifying and verifying dynamic properties of Web sites. We use Maude as a suitable specification language for such rewriting models which also permits to directly explore interesting dynamic properties of Web sites.

Keywords: 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 [2,8,9] have been recently used to reason about the static contents of Web sites [1]. In this paper we show that term rewriting techniques are also well-suited for modeling and reasoning about the dynamic behavior of Web sites. We use Maude [3] 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 Abstract Reduction Systems

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 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 \( \rightarrow_p = \{(p, p_1), \ldots, (p, p_{n_p})\} \subseteq \mathcal{P} \times \mathcal{P} \) which is the abstract relation between the page \( p \) and its immediate successors (i.e., the pages \( p_1, \ldots, p_{n_p} \in \mathcal{P} \) which are reachable from \( p \) in a single step). The pair \( (\mathcal{P}, \rightarrow_p) \), where \( \rightarrow_p = \bigcup_{p \in \mathcal{P}} \rightarrow_p \) is an Abstract Reduction System (ARS [2, Chapter 2]) and we can use the associated computational relations \( \rightarrow_p, \rightarrow_p^+ \), etc., 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 \( p \rightarrow_p^* p' \).

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.

3 Term Rewriting Systems

Term Rewriting Systems (TRSs [2,9]) 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:

\[
p(U) \rightarrow vp(U) \quad vp(u_1) \rightarrow bp(u_1) \quad bp(U) \rightarrow p_1(U)
\]

\[
: \quad : \quad :
\]

\[
vp(u_{mp}) \rightarrow bp(u_{mp}) \quad bp(U) \rightarrow p_{np}(U)
\]

where $vp$ and $bp$ stand for validate and browse page $p$, respectively, and $u_i$ for $1 \leq i \leq mp$ 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\(^3\); thus, reachability is decidable [4]. 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):

\[
\text{mod WebWWV05 is}
\]

\[
\text{sort S .}
\]

\[
\text{ops wwv05 submission speakers org valencia accommodation travelling}
\]

\[
: \quad \rightarrow \quad S .
\]

\[
\text{ops sbmlink entcs entcswwv05 : S \rightarrow S .}
\]

\[
\text{ops login vlogin blogin : S \rightarrow S .}
\]

\[
\text{ops forgotten register submit : S \rightarrow S .}
\]

\[
\text{ops krishnamurthi finkelstein : S \rightarrow S .}
\]

\[
\text{ops alpuente ballis escobar : S \rightarrow S .}
\]

\[
\text{op cfp : \rightarrow S .}
\]

\[
\text{ops slucas smith : \rightarrow S .}
\]

\[
\text{vars P PS X U : S .}
\]

\[
\text{rl wwv05(U) \Rightarrow submission(U) .}
\]

\[
\text{rl wwv05(U) \Rightarrow speakers(U) .}
\]

\[
\text{rl wwv05(U) \Rightarrow org(U) .}
\]

\[
\text{rl wwv05(U) \Rightarrow cfp .}
\]

\[
\text{rl wwv05(U) \Rightarrow valencia(U) .}
\]

\[
\text{rl wwv05(U) \Rightarrow accommodation(U) .}
\]

\[
\text{rl wwv05(U) \Rightarrow travelling(U) .}
\]

\[
\text{rl submission(U) \Rightarrow sbmlink(U) .}
\]

\[
\text{rl submission(U) \Rightarrow entcs(U) .}
\]

\[
\text{rl submission(U) \Rightarrow entcswwv05(U) .}
\]

\[
\text{rl sbmlink(U) \Rightarrow login(U) .}
\]

\[
\text{rl sbmlink(U) \Rightarrow forgotten(U) .}
\]

\[
\text{rl sbmlink(U) \Rightarrow register(U) .}
\]

\[
\text{rl speakers(U) \Rightarrow finkelstein(U) .}
\]

\[
\text{rl speakers(U) \Rightarrow krishnamurthi(U) .}
\]

\[
\text{rl org(U) \Rightarrow alpuente(U) .}
\]

\[
\text{rl org(U) \Rightarrow ballis(U) .}
\]

\[
\text{rl loginslucas(U) \Rightarrow vlogin(U) .}
\]

\[
\text{rl vlogin(slucas) \Rightarrow blogin(slucas) .}
\]

\[
\text{rl blogin(U) \Rightarrow submit(U) .}
\]

\[
\text{endm}
\]

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)

\[^{3}\text{A TRS is shallow if variables occur (at most) at depth 1 both in the left- and right-hand sides of the rules [4, Section 4]. A TRS is linear if variables occur at most once both in left- and right-hand sides of the rules [2, Definition 6.3.1].}\]
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:

\[
\begin{align*}
\text{wwv05(slucas)} & \rightarrow \text{submission(slucas)} \rightarrow \text{sbmlink(slucas)} \\
& \rightarrow \text{login(slucas)} \rightarrow \text{vlogin(slucas)} \rightarrow \text{blogin(slucas)} \\
& \rightarrow \text{submit(slucas)}
\end{align*}
\]

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)

4 Rewriting model and logics

In Software Engineering (also in Web design), we need to have high level languages to specify problems and properties, and efficient algorithms to compute. In term rewriting, this duality has a nice counterpart as logic languages and classes of automata [5,11]. Regarding the application of these ideas to reasoning about Web sites, we discuss a number of interesting properties which can be expressed in different frameworks but which also suggest that further research should be done.

Rewriting theories are first-order logic where the variables of the logic language range on ground terms (i.e., terms which only contain function symbols from the underlying signature) and atomic formulas are of the form \( x \rightarrow_R y \) (one-step rewriting) or \( x \rightarrow^* y \) (many steps rewriting) associated to TRSs \( R \). For instance, the equality is expressible in this logic, since \( x = y \) is the formula \( x \rightarrow^* \emptyset y \) associated to the empty TRS [5]. The property “There are Web pages containing no link” can then be easily represented by the formula:

\[
\exists x \forall y (\neg (x \rightarrow_R y))
\]

where \( R \) represents the Web site. On the other hand, the property “There is no unreachable page” (from the ‘main’ page) could be expressed as follows:

\[
\exists u \left( \bigvee_{1 \leq i \leq m} \text{main}(u) \rightarrow^*_R p_i(u) \lor \bigvee_{1 \leq j \leq n} \text{main}(u) \rightarrow^*_R q_j \right)
\]
where $p_1, \ldots, p_m$ are the monadic symbols used to give name to Web pages in the system and $q_1, \ldots, q_n$ are the names represented as constant symbols. This is not a valid formula, however, in the rewriting theory described above: we need to make symbols explicit, but the symbols in the signatures are not part of the syntax of the logic; moreover, it would only make real sense if $u$ is intended to range on user-names rather than arbitrary terms (i.e., some type/sort discipline would be necessary).

Fortunately, properties referring the structure of terms from a signature, can often be expressed in a second-order logic like the (weak) second-order monadic logic with $k$ successors (WSkS) [10,11]. The rewrite relation $\rightarrow^*_R$ for left-linear and right-ground TRSs $\mathcal{R}$ is definable in WSkS [4]. Although using quite a different notation, the previous sentence could then be rephrased as a WSkS formula.

Another example is the property "Identity changes are not possible" which could be written as follows:

$$\forall u \forall v \left( \bigvee_{1 \leq i \leq m} \text{main}(u) \rightarrow^*_R p_i(v) \right) \Rightarrow u = v$$

Again, this is WSkS-definable (for left-linear and right-ground TRSs $\mathcal{R}$).

Now, the interesting point is deciding the truth or falsity of such properties for a concrete Web site (i.e., TRS) $\mathcal{R}$. For rewriting theories, this is possible if the underlying TRSs $\mathcal{R}$ are ground [6]. In our case, this would be appropriate for the ARS model described in Section 2 (since ARSs are very simple ground TRSs), but it does not apply to the rewriting model in Section 3. Some decidability results are known for more general (yet quite restrictive) classes of TRSs. For instance, the one-step rewriting theory (i.e., only formulas $x \rightarrow^*_R y$ are allowed) is decidable for linear TRSs $\mathcal{R}$ where (shared) variables occur at the same depth both in the left- or right-hand sides of the rules [7, Proposition 178]. Our rewriting model satisfy these syntactic restrictions. However, the obtained framework is too weak: only truth or falsity of sentence (1) above could be proved! The WSkS logic is also decidable [10,11]; unfortunately, though, our rewriting model does not yield left-linear and right-ground TRSs, in general. Hopefully, a more accurate analysis of representability and decidability issues will yield more applicable results (in our setting) if the very simple shape of the rules used in the rewriting model associated to the Web sites is taken into account: note that all symbols are at most monadic and only flat terms of the form $f(x)$ for a monadic symbol $f$ and a variable $x$ eventually occur in either the left- or right-hand sides.
5 Further improvements and applications

The basic model in Section 3 can be improved in a number of different ways to obtain more expressive models and/or analyze other behavioral issues: For instance, quantitative information (dealing with length of paths, frequency of use, . . . ) could be added to the basic model. Evolving Web sites should also be considered: adding new pages to a Web site is quite usual. This corresponds to dynamically adding new rules to the model of the site. This could be modeled by using transformations which preserve concrete invariants (which could be given as sentences of an appropriate logic as discussed above). Also, from a logical point of view, the following questions are interesting:

(i) Which are the appropriate (fragments of) logics which are useful to specify (and reason about) the dynamic behavior of Web sites?

(ii) How types, strategies, conditional rules, etc., can help to get a more expressive model or to improve its power from a logic point of view?

Finally, the rewriting theory could also benefit from the new research directions pointed by the analysis of the Web. For instance, Web sites can often be considered as composed by many smaller sites. This can be connected with the analysis of modular properties in Term Rewriting [8], but the current developments are probably too weak for modeling Web site structures and analyzing the related/relevant properties.

References


