Theoretical Computer Science 11 (1980) TCS 39–47

SPEEDING UP SUBTREE REPLACEMENT SYSTEMS

John STAPLES*

Department of Mathematics and Computer Science, Queensland Institute of Technology, Brisbane 4001, Australia

Communicated by A. Salomaa
Received August 1978
Revised March 1979

Abstract. This paper is the last in a three-part study of efficient evaluations of expressions. Here the optimal evaluation algorithms for systems of graph-like expressions studied earlier are compared with the best possible performance of evaluation algorithms for classical systems of expressions. For a wide class of systems it is shown that a graph-like version can speed up the classical system while maintaining correctness.

1. Introduction

1.1. This paper is the third part of a study begun in [5, 6] of computation on graph-like expressions. Familiarity with [5] is assumed. Here we show that a wide class of subtree replacement systems in the sense of Rosen [3] can be strongly speeded up, in the sense of [4, Section 6.5], by systems of graph-like expressions in the sense of [5].

The subtree replacement systems so speeded up belong to a class which has previously been shown by Rosen [3, Theorem 6.5] to have the Church–Rosser property; see Hindley [2, p. 7] for an alternative description of that class. The description which we shall give here of the subclass which interests us is however chosen for its compatibility with the concept of graph-like expression introduced in [5].

The proof of speedup is basically an application of the abstract speedup theorem of [4, Section 7].

1.2. Throughout this paper all terms and expression graphs considered are acyclic. Expression graphs may be called simply 'graphs'.

* Current address: Department of Computer Science, University of Queensland, St. Lucia 4067, Australia.

2. Subtree replacement systems

2.1. Our first task is to describe the subtree replacement systems of Rosen [3] in the notation of [5]. To begin with, a term T in the sense of Section 1.2 is called a *tree* if it satisfies the following two conditions:

(i) every address of the term other than the root occurs in just one value of the term, and just once in that value,

(ii) for every assignment $s := v$ of the term, s is a singleton set and v is not the empty value.

2.2. A *tree rule* is defined to be a replacement rule

$$\rho = (R_\rho, C_\rho)$$

as defined in [5] such that R_ρ, C_ρ are trees; say with roots r_ρ, c_ρ respectively.

2.3. A *subtree replacement system* is a system in the sense of [6, Section 1.2] such that the terms of the system are trees and the contractions of the system are all instances of some fixed set of tree rules.

2.4. In order to describe the subtree replacement systems which are to be speeded up we next define a *tree rule scheme* to be a replacement rule $\rho = (R_\rho, C_\rho)$ which satisfies the following conditions:

(i) every address of R_ρ (respectively C_ρ) except its root occurs in just one value of the term, and just once in that value,

(ii) for every assignment $s := v$ of R_ρ, s is a singleton set,

(iii) for every assignment $s := v$ of C_ρ such that v is not the empty value, s is a singleton set,

(iv) every empty-valued address of C_ρ is a committed address of the rule.

Note that it is essential for us that distinct empty-valued addresses of C_ρ may be equivalent. Empty-valued addresses play in the present approach the role of the parameters of Rosen [3, Section 6], called meta-variables by Hindley [2]; equivalent empty-valued addresses of C_ρ correspond to occurrences of the same parameter.

2.5. For C_ρ as in Section 2.4 we write C_ρ^{T} for the term which is obtained from C_ρ by making inequivalent addresses which are equivalent in C_ρ.

2.6. A tree rule $\sigma = (R_\sigma, C_\sigma)$ is said to *match* a tree rule scheme ρ as in Section 2.4 if there is a hom $h : R_\rho \to R_\sigma$ and a hom $h' : C_\rho^{\mathrm{T}} \to C_\sigma$ such that h, h' map the roots of R_ρ, C_ρ^{T} to the roots of R_σ, C_σ respectively and for every empty-valued address $\boldsymbol{a}$ in C_ρ, $h'(\boldsymbol{a})$ is the root of a subtree of C_σ which is isomorphic to the subtree of R_σ which has root $h(a)$.

2.7. A set D of tree rule schemes is called *disjoint* if for all terms T, all $\rho, \sigma \in D$ and all instances $h: R_\rho \to T$, $k: R_\sigma \to T$ of ρ, σ respectively applied to T; either $\rho = \sigma$ and $h = k$, or else h and k are disjoint.

2.8. A subtree replacement system $(V, \Rightarrow_t)$ is called *quasidisjoint* if there is some disjoint set D of tree rule schemes such that the set D^T of tree rules which defines $\Rightarrow_t$ is the set of tree rules which match some scheme of D.

2.9. In order to describe the subcommutative systems which will be used to speed up quasidisjoint subtree replacement systems we define a *covering* $\gamma: T \to U$ to be a hom γ which maps the root of T to the root of U and which maps empty-valued addresses to empty-valued addresses.

Evidently,

Property 1. *The composition of two coverings is a covering.*

2.10. Given a set V of trees, we write V^+ for the set of all terms T^+ such that for some $T \in V$ there is a covering $T \to T^+$.

2.11. Our main result will be that, in the notation of Section 2.8 and 2.11, every quasidisjoint subtree replacement system $(V, \Rightarrow_t)$ can be strongly speeded up, in the sense of [4], by the subcommutative system $(V^+, \Rightarrow_g)$, where $\Rightarrow_g$ is the set of all contractions of terms of V^+ which are defined by instances of elements of D.

2.12. We conclude this section by indicating that quasidisjoint subtree replacement systems satisfy the hypotheses of Rosen [3, Theorem 6.5]; we discuss those hypotheses in the form given by Hindley [2]. In Hindley's notation the hypotheses are (Ci), $i = 1, \ldots, 5$, where:

- (C1) is just closure under the application of the rules, which is implied by our assumption that $(V, \Rightarrow_t)$ is a subtree replacement system,
- (C2) is given by our assumption of disjointness,
- (C3) is given by (iv) of the definition of tree rule scheme,
- (C4) is given by (ii) of the definition of tree rule scheme,
- (C5) is given by the disjointness assumption.

3. The speedup result

3.1. Given a quasidisjoint subtree replacement system $\mathscr{B} = (V, \Rightarrow_t)$ as in Section 2.8, we show that the set V^+ defined in Section 2.11 is closed under the application of instances of the disjoint set D of rules which defines $\Rightarrow_g$. Thus $\mathscr{C} = (V^+, \Rightarrow_g)$ is subcommutative, from [5]. We also show that $\mathscr{C}$ strongly speeds up $\mathscr{B}$ in the sense of [4, Section 6.5], basically by applying the abstract speedup Theorem 7.6 of [4] to prove weak speedup.

3.2. In order to apply that abstract speedup theorem we define as follows a relation $S \subseteq V \times V^+$ and an ordering $\geqslant$ on each set

$$S(b) = \{c : (b, c) \in S\}, \quad b \in V,$$

in such a way that for each b with normal form in V, $S(b)$ is nonempty and has a greatest element which we shall denote $g(b)$.

We define $(b, c) \in S$ if and only if $b \in V$, $c \in V^+$ and there is a covering $b \to c$. For $(b, c_1), (b, c_2) \in S$ we define $c_1 \geqslant c_2$ if and only if there is a covering $c_1 \to c_2$. It is then elementary that for all $b \in V$, $S(b)$ is nonempty, and that $\geqslant$ is a partial order which has as its greatest element b itself. It is also elementary from this definition that, in the notation of [4, Section 7.6], if S_N is a weak speedup of $\mathscr{B}$ by $\mathscr{C}$, then it is also a strong speedup; since for every term U there is at most one tree T such that there is a covering $T \to U$.

3.3. The following lemma, which verifies condition (iv) of the abstract speedup theorem, is evident from the definition of $\Rightarrow_t$.

Lemma 1. *For all $a, b \in V$ such that $a \Rightarrow_t b$ there is $c \in V^+$ and a covering $b \to c$ such that $a \Rightarrow_g c$.*

It is also evident that, in satisfaction of condition (ii) of the abstract speedup theorem,

Lemma 2. *For all $b \in V$ and all coverings $b \to c$, b is in normal form in $\mathscr{B}$ if and only if c is in normal form in $\mathscr{C}$.*

3.4. In order to verify the remaining conditions (i) and (iii) of the abstract speedup theorem, as well as to complete the proof that V^+ is closed under contraction, we introduce a notion for terms which corresponds to a particular case of a concept which is wellknown in the theory of subtree replacement systems.

First we say that a set $\{a_1, \ldots, a_k\}$, $k \geqslant 1$, of addresses of a term T is a *special* set if there are paths $p_1, \ldots, p_k$ which all start at the root of T and are such that p_i has end a_i, and no address a_j is equivalent to an address of p_i, $j \neq i$, $i, j = 1, \ldots, k$.

We recall from [6, Lemma 2] that

Lemma 3. *If $h : R_\rho \to T$ is an instance of a rule ρ and if p is a path of T which starts at the root of T and does not include any address equivalent to $h(r_\rho)$, the image under h of* *the root of R_ρ, then p is also a path in the term U which is obtained by contracting T according to h.*

Hence, in the notation of Section 3.6 and 3.7:

Lemma 4. *If $\{h(r_\rho), a_1, a_2\}$ is a special set of addresses of T, then $\{a_1, a_2\}$ is a special set of addresses of U.*

3.5. Next we define a set $\{h_1, \ldots, h_k\}$, $k \geq 1$, of disjoint instances of a rule ρ applied to a term T to be a *special set* of instances of ρ, if $\{h_1(r_\rho), \ldots, h_k(r_\rho)\}$ is a special set of addresses of T. It follows from Lemma 4 that:

Result 1. *If $\{h_1, \ldots, h_k\}$ is a special set of instances $R_\rho \to T$ and if h_1 defines the contraction $T \Rightarrow U$, then $\{h_2, \ldots, h_k\}$ is a special set of instances $R_\rho \to U$.*

Thus the subcommutativity result of [5] gives the following commutativity result:

Result 2. *If $\{h_1, \ldots, h_k\}$ is a special set of instances $R_\rho \to T$, then those instances can be applied in arbitrary order to give reductions of length k which all reduce T to the same term U.*

3.6. We call a *special reduction* any reduction which is, as in Result 1, defined by a special set of instances.

The following lemma is useful for the application of the notion of special reduction:

Lemma 5. *For every tree rule scheme ρ, every instance $h: R_\rho \to U$, every covering $\gamma: T \to U$ and every address a of T such that $\gamma(a) = h(r_\rho)$, there is a unique instance $h': R_\rho \to T$ of ρ such that*

$$h'(r_\rho) = a.$$

Proof. Define $h'(p)$ for $p \in R_\rho$ by induction on the distance of the shortest path from r_ρ to p, and simultaneously check that whenever p has value of the form $f(p_1, \ldots, p_m)$, $m \geq 0$, then $h'(p)$ has value of the form $f(q_1, \ldots, q_m)$, where $\gamma(q_i)$, $h(p_i)$ are equivalent, $i = 1, \ldots, m$.

First, define $h'(r_\rho) = a$. In the case that r_ρ has value of the form $f(p_1, \ldots, p_m)$, $h(r_\rho)$ has value of the form $f(s_1, \ldots, s_m)$ where s_i is equivalent to $h(p_i)$. Thus as $\gamma(a) = h(r_\rho)$ and γ is a covering, a has value of the form $f(q_1, \ldots, q_m)$, where $\gamma(q_i)$ and s_i are equivalent, $i = 1, \ldots, m$; so that $\gamma(q_i)$ and $h(p_i)$ are equivalent as required.

Next suppose that the length l of the shortest path from r_ρ to an address p is positive, and that $h'(p')$ is defined for addresses p' at shorter distance from r_ρ. Now p

occurs as, say, the ith address of some value, $g(t_1, \ldots, t_n)$ say, of some address p' for which $h'(p')$ has already been defined; and p occurs in this way once only, since ρ is a tree rule scheme. By inductive hypothesis, $h'(p')$ has value of the form $g(u_1, \ldots, u_n)$ where $\gamma(u_i)$ and $h(p_i)$ are equivalent; we define $h'(p)$ to be u_i. If then p has value of the form $f(p_1, \ldots, p_m)$, then as γ is a covering we find as before that $h'(p)$ has value $f(q_1, \ldots, q_m)$, where $\gamma(q_i)$, $h(p_i)$ are equivalent, $i = 1, \ldots, m$.

3.7. We next state two lemmas which enable the proof of the main result to be concluded. The proof of the main result from these lemmas will then be given; finally we shall return to the proofs of the lemmas.

Lemma 6. *For every tree $b \in V$ and every special reduction $b \Rightarrow_g^* c$ there is a reduction $b \Rightarrow_t^* b'$ in $\mathcal{B}$ such that there is a covering $b' \to c$ (see Fig.* 1).

$$\begin{tikzcd} & & b' \arrow[d, dashed, "\text{covering}"] \\ b \arrow[urr, dashed, "\mathcal{B}\text{-reduction}"] \arrow[rr, "\text{special reduction}"'] & & c \end{tikzcd}$$

Fig. 1.

Proof. See Section 3.

Lemma 7. *For all $c_1, c_2 \in V^+$, for every covering $\gamma : c_1 \to c_2$ and for every contraction $c_2 \Rightarrow_g e_2$ there is a special reduction $c_1 \Rightarrow_g^* e_1$ such that there is a covering $\delta : e_1 \to e_2$ (see Fig.* 2).

Proof. See Section 3.

$$\begin{tikzcd} c_1 \arrow[r, dashed, "\text{special reduction}"] \arrow[d, "\gamma"'] & e_1 \arrow[d, dashed, "\delta"] \\ c_2 \arrow[r, "g"'] & e_2 \end{tikzcd}$$

Fig. 2.

3.8. Our first application of Lemma 7 is the following result, which will verify condition (iii) of the abstract speedup theorem, once we have shown that V^+ is closed under contraction by instances of rules of D.

Result 3. *For all $c_1, c_2 \in V^+$ such that there is a covering $\gamma : c_1 \to c_2$ and for every contraction $c_1 \Rightarrow_g c'$ there is a contraction $c_2 \Rightarrow_g e_2$ and a reduction $c' \Rightarrow_g^* e_1$ such that there is a covering $e_1 \to e_2$ (see Fig. 3).*

$$\begin{tikzcd}
c_1 \arrow[r, "+"] \arrow[d, "\delta"'] & c' \arrow[r, dashed, "+\text{reduction}"] & e_1 \arrow[d, dashed, "\text{covering}"] \\
c_2 \arrow[rr, dashed, "+"'] & & e_2
\end{tikzcd}$$

Fig. 3.

Proof. Say $c_1 \Rightarrow_g c'$ is defined by an instance $h : R_\rho \to c_1$ of a rule ρ. Then $\gamma \circ h$ is an instance $R_\rho \to c_2$ of ρ, which defines a contraction $c_2 \Rightarrow_g e_2$ say. From Lemma 7 there is an induced special reduction $c_1 \Rightarrow_g^* e_1$ such that there is a covering $e_1 \to e_2$. Evidently h is one of the instances of the special set which defines the special reduction $c_1 \Rightarrow_g^* e_1$, and we can choose to perform this special reduction by first performing the contraction defined by h, thus giving the required reduction $c_1 \Rightarrow_g c' \Rightarrow_g^* e_1$.

3.9. The following combination of Lemma 6 and 7 shows that V^+ is closed under contraction and that condition (i) of the abstract speedup theorem is satisfied. It therefore concludes the proof of the result of Section 3.1, apart from the proofs of Lemma 6 and 7.

Result 4. *For all $b \in V$, for all coverings $b \to c$ and for all contractions $c \Rightarrow_g e_2$ which are defined by the rules of D, there is a reduction $b \Rightarrow_t^* b'$ such that there is a covering $\delta : b' \to e_2$ (see Fig. 4).*

$$\begin{tikzcd}
b \arrow[r, dashed, "*", "t"'] \arrow[d, "\text{covering}"'] & b' \arrow[d, dashed, "\text{covering } \delta"] \\
c \arrow[r, "g"'] & e_2
\end{tikzcd}$$

Fig. 4.

Proof. From Lemma 7 there is a special reduction $b \Rightarrow_g^* e_1$ such that there is a covering $e_1 \to e_2$. From Lemma 6 there is therefore a reduction $b \Rightarrow_t^* b'$ in $\mathscr{B}$ such that there is a covering $b' \to e_1$. Since the composition of two coverings is a covering, we have the stated result.

3.10. *Proof of Lemma 6*

Write $h : R_\rho \to b$ for the instance of tree rule scheme ρ which defines $b \Rightarrow_g c$. Note that all assignments of R_ρ and C_ρ with nonempty values have unique addresses, and all addresses of R_ρ, C_ρ except the root occur in some value. Thus the only difference between the action $b \Rightarrow_g c$ of h and the action, $b \Rightarrow_t b'$ say, of the corresponding tree rule instance is merely that some identical disjoint subtrees of the tree b' may be identified to form c; thus Lemma 6 is evident.

3.11. *Proof of Lemma 7*

Write $h : R_\rho \to c_2$ for the instance of the rule ρ of D which defines the contraction $c_2 \Rightarrow_g e_2$. We first show that the set $\gamma^{-1}(h(r_\rho))$ is a special set of addresses of c_1. To see that, notice that the image of every path under the hom γ is again a path, of the same length. Thus if the set $\gamma^{-1}(h(r_\rho))$ had two elements a_1 and a_2, $a_1 \neq a_2$, such that a_2 occurred as an address on some path from the root of c_1 to a_2, then there would be a path in c_1 from a_1 to a_2 of positive length, and hence a path in c_2 of positive length from $\gamma(a_1) = h(r_\rho)$ to $\gamma(a_2) = h(r_\rho)$. That is, c_2 would include a cycle, contradiction.

Notice next that, as γ is a covering, for each address a of $\gamma^{-1}(h(r_\rho))$ there is an instance $h_a : R_\rho \to c_1$ of ρ such that $a = h_a(r_\rho)$; and the instances defined by distinct elements of $\gamma^{-1}(h(r_\rho))$ are disjoint since D is disjoint. Thus these instances form a special set which defines a special reduction, $c_1 \Rightarrow_g^* e_1$ say. It remains to show that there is a covering $\delta : e_1 \to e_2$, which makes the diagram of Fig. 2 commute.

To do that we first write $c_2 \Rightarrow f_2 \Rightarrow e_2$ for the contraction $c_2 \Rightarrow e_2$, where as in [5, Section 2.11 (iv)], f_2 is the set of assignments from which e_2 is obtained by deleting addresses which are not on a path from the root of c_2. It is enough to show that there is a map $\delta : e_1 \to f_2$ with the properties:

(i) δ maps $h_a(r_\rho)$ to $h(r_\rho)$,

(ii) images of empty-valued addresses of e_1 are empty-valued addresses of f_2,

(iii) δ preserves assignments. That is,

(1) for every assignment $s := v$ of e_1 there is an assignment $s' := v'$ of f_2 such that $\delta(s) \subseteq s'$,

(2) for every assignment $s := f(p_1, \ldots, p_m)$ of e_1, and $s' := v'$ in f_2 such that $\delta(s) \subseteq s'$, v' has the form $f(q_1, \ldots, q_m)$, where $\delta(p_i)$ and q_i are equivalent.

But such a map δ is evidently defined as follows:

Case 1. p is an address of c_1. Then define $\delta(p)$ to be p.

Case 2. p is an address of an introduced copy of C. Then define $\delta(p)$ to be the corresponding address of the copy of C which was introduced into c_2 to produce f_2.

4. Discussion

Quasidisjointness is an evident property of the more elementary examples of subtree replacement systems, such as weak combinatory logic and the systems which arise in the study of McCarthy's recursive definitions. However the usual form of the lambda calculus is not quasidisjoint.

References

[1] H. Ehrig and B.K. Rosen, Commutativity of independent transformations on complex objects, I.B.M. Research Report RC 6251 (1976).
[2] J.R. Hindley, An abstract Church–Rosser theorem. II: applications, *J. Symbolic Logic* **39** (1974) 1–21.
[3] B.K. Rosen, Tree-manipulating systems and Church–Rosser theorems, *JACM* **20** (1973) 160–187.
[4] J. Staples, A class of replacement systems with simple optimality theory, *Bull. Austral. Math. Soc.* **17** (1977) 335–350.
[5] J. Staples, Computation on graph-like expressions, *Theoret. Comput. Sci.* **10** (1980) 171–185.
[6] J. Staples, Optimal evaluations of graph-like expressions, *Theoret. Comput. Sci.* **10** (1980) 297–316.