Computing flat vectorial Boolean fixed points
Preliminary draft notes

Damian Niwiński*

November 6, 2006

The goal of this note is to show that, for a fixed k, the value of a closed vectorial Boolean fixed-point term

$$\theta_1 \mathbf{x_1}.\theta_2 \mathbf{x_2} \ldots \theta_{\mathbf{n}} \mathbf{x_n}.\tau$$

can be computed in polynomial time. Here we assume that $\tau = (\tau_1, \ldots, \tau_k)$, where each τ_i is built from variables, disjunction and conjunction, but otherwise need not be in any normal form.

Vector expressions

We fix a countable set of conveniently indexed variables

$$V \quad = \quad \{x_{n,m} : 1 \leq n < \omega, 1 \leq m \leq k\}.$$

The set of terms over a set of variables $Y \subseteq V$, is defined by the following clauses:

- $Y \subseteq T(Y)$,
- for each *finite* $Q \subseteq T(Y)$, $\bigwedge Q$ and $\bigvee Q$ are terms in $T(Y)$, we abbreviate $\bigvee \emptyset = \top$, and $\bigwedge \emptyset = \bot$.

We are interested in evaluating expressions of the form

$$\Gamma = \theta_1 \begin{pmatrix} x_{1,1} \\ \vdots \\ x_{1,k} \end{pmatrix} . \theta_2 \begin{pmatrix} x_{2,1} \\ \vdots \\ x_{2,k} \end{pmatrix} . \cdots . \theta_n \begin{pmatrix} x_{n,1} \\ \vdots \\ x_{n,k} \end{pmatrix} . \begin{pmatrix} \tau_1 \\ \vdots \\ \tau_k \end{pmatrix}$$

where $\theta_i \in \{\mu, \nu\}$, and each τ_j is a term in $T(V_n)$, where, generally

$$V_\ell \quad = \quad \{x_{i,j} : 1 \leq i \leq \ell, \, 1 \leq j \leq k\},$$

(thus $V_0 = \emptyset$). In the sequel we usually abbreviate

$$\Gamma \quad = \quad \theta_1 \mathbf{x_1}.\theta_2 \mathbf{x_2} \ldots \theta_n \mathbf{x_n}.\tau$$

We interpret such expressions over the Boolean algebra $\{0, 1\}$. Each term $t \in T(V_n)$ is interpreted as a mapping on the set of valuations, $[\![t]\!] : \{0,1\}^{V_n} \to \{0,1\}$, in the usual manner. Hence τ induces a mapping

$$[\![\tau]\!] : \quad \{0,1\}^{V_n} \ni v \quad \mapsto \quad ([\![\tau_1]\!]v, \ldots, [\![\tau_k]\!]v) \in \{0,1\}^k$$

To proceed, it is convenient to use an abbreviation

$$\Gamma_i = \theta_{i+1} \mathbf{x_{i+1}} . \ldots \theta_n \mathbf{x_n} . \tau$$

*Note written while visiting the Newton Institute, Cambridge, in June 2006.

Thus $\Gamma_n = \tau$. Now suppose we have already defined

$$[\![\Gamma_i]\!] : \{0,1\}^{V_i} \to \{0,1\}^k.$$

For any $w \in \{0,1\}^{V_{i-1}}$, consider the mapping $\{0,1\}^k \to \{0,1\}^k$ given by

$$\lambda \mathbf{a} \in \{0,1\}^k . [\![\Gamma_i]\!] w[\mathbf{a}],$$

where $w[\mathbf{a}]$ is a valuation in $\{0,1\}^{V_i}$ given by

$$\begin{array}{rcll} w[\mathbf{a}](x_{i,j}) & = & a_j & \text{for } j = 1, \ldots, k \\ w[\mathbf{a}](x_{i',j}) & = & w(x_{i',j}) & \text{for } i' < i. \end{array}$$

By construction, this mapping is monotone, so it has the least fixed point $\mu\mathbf{a}.[\![\Gamma_i]\!] w[\mathbf{a}]$, and the greatest fixed point $\nu\mathbf{a}.[\![\Gamma_i]\!] w[\mathbf{a}]$. If $\theta_i = \mu$, we let

$$[\![\Gamma_{i-1}]\!] : \quad \{0,1\}^{V_{i-1}} \ni v \quad \mapsto \quad \mu\mathbf{a}.[\![\Gamma_i]\!] v[\mathbf{a}].$$

If $\theta_i = \nu$, the definition is analogous with ν replacing μ.

For $i = 0$, we finally obtain

$$[\![\Gamma_0]\!] = [\![\Gamma]\!] \in \{0,1\}^k.$$

Note It follows from the general properties of fixed points (see, e.g., [2]) that $\ldots \theta x . \theta x' . \ldots . \varphi(\ldots x \ldots x' \ldots)$ can be reduced to $\ldots \theta x . \ldots . \varphi(\ldots x \ldots x \ldots)$, hence it is enough to consider expressions where the fixed point operators alternate.

Remark Let M be a set of k elements, say $M = \{1, \ldots, k\}$. Then it is not difficult to see that any monotone mapping $F : (\wp M)^n \to \wp M$ can be represented by a a vectorial term τ in $T(V_n)$ in such a way that, for any $A_1, \ldots, A_n \subseteq M$, and $i \in M$,

$$i \in F(\mathbf{A}) \iff [\![\tau_i]\!] v_{\mathbf{A}} = 1,$$

where $v_{\mathbf{A}}$ is a valuation defined by

$$v_{\mathbf{A}}(x_{i,j}) \quad = \quad 1 \quad \text{iff} \quad j \in A_i.$$

Games

We assume the reader is familiar with parity games. For Γ as above, we fix a function $\widehat{rank} : \{1, \ldots, n\} \to \omega$, such that

- $\widehat{rank}(i)$ is even iff $\theta_i = \nu$;
- if $i < i'$ (i.e., θ_i precedes $\theta_{i'}$ then $\widehat{rank}(i) \geq \widehat{rank}(i')$.

We may assume that $\widehat{rank}$ is chosen with minimal range.

We define the game $G(\Gamma)$ as follows.

- The set of positions consists of all subterms of $\tau_1, \ldots, \tau_k$.
- Positions of Eve are terms of the form $\bigvee Q$, and (for concreteness) variables.

  The remaining (i.e., conjunctions $\bigwedge Q$) are positions of Adam.
- The moves $p \to p'$ are of the following kinds:
  - $(\bigvee Q) \to t$, and $(\bigwedge Q) \to t$, whenever $t \in Q$,

– $x_{i,j} \to \tau_j$.

- The ranking function is given by

$$\begin{array}{rcll} rank(x_{i,j}) & = & \widehat{rank(i)} & \\ rank(t) & = & 0 & \text{for composed terms} \end{array}$$

As usual, we assume that Eve wins an infinite play if the *highest* rank occurring infinitely often is *even*. Note that positions $\top$ and $\bot$ are terminating, and the respective player looses, as he (or she) cannot make a move.

The connection between this game and the semantics of Γ is given by the following. Let $[\![\Gamma]\!] = \mathbf{g} = (g_1, \ldots, g_k)$, and let $v_{\mathbf{g}}$ denote the valuation given by

$$v_{\mathbf{g}}(x_{i,j}) = g_j, \quad \text{for } j = 1, \ldots, k.$$

Proposition 1 *A position t is winning for Eve in $G(\Gamma)$ if and only if $[\![t]\!]v_{\mathbf{g}} = 1$.*

The above fact is at the basis of the connection between the μ-calculus model checking and parity games. However it may have not appeared in the literature precisely in this form. In [2], it is explicitly stated and proved for positions of the form $x_{i,j}$, assuming that terms τ_j are in disjunctive normal form (Proposition 4.4.2, page 95). Using the technique of expanding the vector[1], it is routine to extend it to the desired claim.

Algorithm

Let us first recall that there is an algorithm which solves parity games of *one player* in polynomial time (in fact even in time $\mathcal{O}(n \cdot \log n)$, [3]). It implies that within this complexity bound we can compute $[\![\Gamma]\!]$ if each τ_i is formed using only conjunction (or using only disjunction).

This can be further extended to the case when each τ_i is in disjunctive normal form (DNF), i.e., a disjunction of conjunctions. The algorithm checks all positional strategies for Eve by the exhaustive search, that is, for each $i = 1, \ldots, k$, selects one disjunct of τ_i, and then computes the value of the resulted vectorial term (which now contains only conjunctions). The correctness follows form the positional determinacy of parity games or, equivalently, from the *selection property* of the vectorial μ-calculus [1] (see also [2]). Note that there is $\mathcal{O}(n^k)$ possibilities of such choice (where n stands for the global size of Γ). So the total time of computing $[\![\Gamma]\!]$ is

$$\mathcal{O}(n^k) \cdot \mathcal{O}(n \cdot \log n) = \mathcal{O}(n^{k+1} \cdot \log n)$$

Note that this complexity does not depend on the number of alternations between fixed-point operators, which is the case of most fixed-point algorithms.

Now let Γ be arbitrary, as at the beginning of this note, but we assume k being fixed. Of course, transforming Γ to DNF would be too expensive. We overcome this, however, by subsequent elimination of redundant variables.

It is useful to have the following ordering on natural numbers:

$$k \sqsubseteq \ell \quad \Leftrightarrow \quad (-1)^k \cdot k \leq (-1)^\ell \cdot \ell.$$

This is the "goodness" ordering for Eve. The crucial observation is that if Adam moves from a conjunction where both $x_{i,j}$ and $x_{i',j}$ appear and he wishes that the next position be τ_j then he will prefer $x_{i,j}$ to $x_{i',j}$, whenever $i \sqsubset i'$.

[1] Intuitively, in terms of systems of equations, it can be explained as replacing equation $x = s(\ldots t \ldots)$ by two new equations

$$\begin{array}{rcl} x & = & s(\ldots z \ldots) \\ z & = & t \end{array}$$

where z is a fresh variable.

Definition A *short conjunction* is a term of the form $\bigwedge Q$, where $Q \subseteq V_n$, and, for each $j = 1, \ldots, k$, there is *at most* one variable $x_{i,j} \in Q$.

Let $c = \bigwedge P$ be an arbitrary conjunction with $P \subseteq V_n$. We let $A(c)$ be the short conjunction $\bigwedge Q$, where $Q \subseteq P$, and, whenever $x_{i',j} \in P$, there is $x_{i,j} \in Q$, with $i \sqsubseteq i'$. (That is, for each j, we leave only one representative $x_{i,j}$, with $\sqsubseteq$-minimal i.)

Note that a short conjunction has at most k variables, and hence there are $\mathcal{O}(n^k)$ possible short conjunctions.

Now suppose an expression

$$\Gamma \quad = \quad \theta_1 \mathbf{x_1}.\theta_2 \mathbf{x_2} \ldots \theta_n \mathbf{x_n}.\tau$$

is given. Our algorithm gradually transforms each τ_i into a term τ_i' in disjunctive normal form, where moreover each conjunction is short. The terms τ_i and τ_i' need not be equivalent, but nevertheless, the resulting vectorial term

$$\Gamma' \quad = \quad \theta_1 \mathbf{x_1}.\theta_2 \mathbf{x_2} \ldots \theta_n \mathbf{x_n}.\,(\tau_1', \ldots, \tau_k')$$

will be equivalent to Γ.

Lemma 1 *Suppose $\bigwedge P$ occurs as a subterm in some τ_i, and let $\Gamma[A(\bigwedge P)]$ denote the vectorial term obtained from Γ by replacing this particular occurrence of $\bigwedge P$ in τ_i by $A(\bigwedge P)$. Then the semantics does not change, i.e.,*

$$[\![\Gamma]\!] \quad = \quad [\![\Gamma[A(\bigwedge P)]]\!]$$

The claim follows easily from Proposition 1, if we analyse possible moves of Adam from the positions $\bigwedge P$ and $A(\bigwedge P)$ in respective games.

The algorithm proceeds in bottom-up fashion. At the first step, we transform all innermost conjunctions in terms τ_i into short ones, using Lemma 1. Consequently, whenever φ in DNF is a subterm of τ_i, all conjunctions in φ are short.

If $\varphi = \tau_i$, we are done.

Otherwise, τ_i contains a subterm $\bigwedge Q$, where each $\varphi \in Q$ is in DNF. We then transform it into an equivalent term in DNF, but at each step we shorten the conjunctions according to Lemma 1. This can be done in time $|Q| \cdot (k^{\mathcal{O}(1)} \cdot n^{2k})$. The case of subterms of the form $\bigvee Q$, where each $\varphi \in Q$ is in DNF, is easy; it is enough to eliminate redundant conjunctions. The total number of the steps of both kind (i.e., for $\bigwedge Q$ or $\bigvee Q$) is proportional to $|\Gamma|$, so that the whole process of transforming Γ to Γ' can be accomplished in polynomial time.

References

[1] A. Arnold. A selection property of the boolean mu-calculus and some of its applications. R.A.I.R.O. Informatique Théorique et Applications, 1997, vol. 31, pp. 371-384.

[2] A. Arnold and D. Niwiński. Rudiments of μ-Calculus. Elsevier Science, Studies in Logic and the Foundations of Mathematics, 146, North–Holland, Amsterdam, 2001.

[3] V. King and O. Kupferman and M.Y. Vardi. On the complexity of parity word automata. Proc. 4th FoSSaCS, LNCS 2030, 2001, pp. 276–286.