MPRI 2-9-1: Well-Quasi-Orders for Algorithms

Final Exam

Documents and computers are forbidden.

Friday, November 22nd 2019

This exam consists of 3 independent parts. When answering the questions, be rigourous in form and complete in reasoning. Rigour does not mean length : you can omit trivial justifications (but do not omit corner cases !). Most questions can be satisfactorily answered in a few lines.

I. Counters with divisibility tests. For $a, b \in \mathbb{N}$, we write "$a \mid b$" when a divides b, that is, when $a > 0$ and $b = ar$ for some $r \in \mathbb{N}$. We write $a \nmid b$ when a does not divide b. Note that 0 divides no number, not even itself.

We consider counter systems with divisibility tests. Such a system has the usual form $S = \langle Loc, C, \Delta\rangle$ with $\Delta \subseteq Loc \times OP(C) \times Loc$. Here the instruction set $OP(C)$ is given by the following abstract grammar :

$$\begin{array}{rcccl} OP(C) \ni op ::= & \texttt{c++} & \mid & \texttt{c--} & \text{/* increments and decrements */} \\ \mid & a \mid \texttt{c}? & \mid & a \nmid \texttt{c}? & \text{/* type 1 tests, positive and negative */} \\ \mid & \texttt{c} \mid a? & \mid & \texttt{c} \nmid a? & \text{/* type 2 tests, positive and negative */} \end{array}$$

where $\texttt{c}$ is any counter from C and a is any constant from $\mathbb{N}$. Recall that decrementing $\texttt{c}$ is only allowed when $\texttt{c}$ contains a strictly positive value, and note that zero tests are not allowed.

1. Give a formal definition of when the relation $\sigma \xrightarrow{\delta} \sigma'$ holds, where σ, σ' are two configurations of S and $\delta = (q, op, q')$ is a transition rule of Δ. *(The point is to check that you have a correct understanding of the operational semantics of the counter systems at hand. Your answer will also establish notation for the following questions.)*
2. Prove that Termination is undecidable for counter systems with divisibility tests. *(Termination asks if a given system S has no infinite run from a given initial configuration σ_0.)*

Solution:
The sequence of three instructions "$\texttt{c++;c} \mid \texttt{1?;c--}$" implements a zero test on $\texttt{c}$. Thus counter systems with type 2 divisibility tests can simulate the Turing-powerful Minsky machines. Hence undecidability.

3. Using WSTS theory, show that termination is decidable for counter systems where only type 1 tests are allowed. *(Do not give an algorithm. Instead, rely on the generic results seen during the course : give a precise definition of the ordering you introduce, list the properties you claim are satisfied, and prove these properties.)*

Solution:
We invoke Prop. 1.36 from the lecture notes. For this we must exhibit a wqo on $Conf_S$ and check the required assumptions.

For $a \in \mathbb{N}$, we define $\leq_a$ by "$x \leq_a y \overset{\text{def}}{\iff} x \leq y \wedge x \equiv y \mod a$" when $a > 0$, and let $\leq_0$ coincide with $\leq$. Now $(\mathbb{N}, \leq_a)$ is a wqo for any $a > 0$. We extend to $\leq_A = \bigcap_{a\in A} \leq_a = \leq_{lcm(A)}$ when A is a finite set of constants, and to $(Conf_S, \leq_S)$, given by

$$\sigma = (q, x_1, \ldots, x_n) \leq_S \rho = (q', y_1, \ldots, y_n) \overset{\text{def}}{\iff} q = q' \wedge x_1 \leq_{A_S} y_1 \wedge \cdots \wedge x_n \leq_{A_S} y_n ,$$

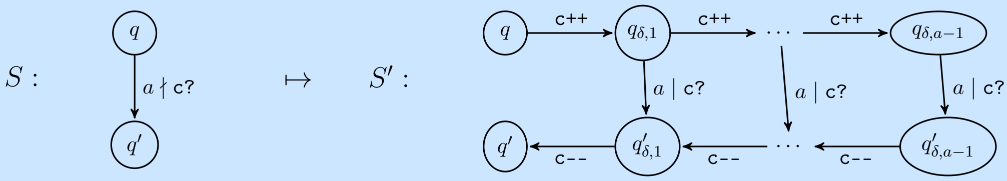
where A_S is the set of constants that occur in Δ. Since A_S is finite, $\leq_{A_S}$ and then $\leq_S$ is a wqo.

The next step is to prove monotonicity of transitions. So assume $\sigma = (q, x_1, \ldots, x_n) \leq_S \rho = (q, y_1, \ldots, y_n)$ and $\sigma \xrightarrow{\delta} \sigma'$ for $\delta = (q, op, q')$. The most interesting case is when op is some $a \mid \mathtt{c}_i?$. Then $\sigma' = (q', x_1, \ldots, x_n)$ and a divides x_i. Now $\sigma \leq_S \rho$ entails $y_i \equiv x_i \mod a$, hence a divides y_i too. There is thus a step $\rho \xrightarrow{\delta} \rho' = (q, y_1, \ldots, y_n)$. Obviously $\sigma' \leq_S \rho'$. The case where op is a negative type 1 test works exactly the same. In the case where op is a decrement or increment, we reuse the reasoning for monotonicity in simple counter systems, with the further check that equality modulo A_S is preserved.

To finish the decidability proof, it remains to check that S is finitely branching (obvious), that $\leq_S$ is decidable (clear), and that one can effectively list the immediate successors of any configuration $\sigma \in \mathit{Conf}_S$ (easy).

4. Give a reduction from counter systems with only type 1 divisibility tests to counter systems with only positive type 1 divisibility tests that preserves termination and reachability. How do you estimate the complexity of your reduction ? *(Precisely, this asks for a reduction $S \mapsto S'$ such that S terminates from some σ_0 iff S' does, and such that some σ' is reachable from some σ in S iff it is in S'.)*

Solution:
We replace any rule $\delta = (q, a \nmid \mathtt{c}?q')$ in S, where $a \geq 2$, with several rules as depicted :



Correctness is clear. Trivial tests $1 \nmid \mathtt{c}?$ are just discarded, and $0 \nmid \mathtt{c}?$ tests are replaced by a no-op, or by a sequence c++; c--.

Regarding complexity, we replace each type 1 negative test "$a \nmid \mathtt{c}?$" with $2a-1$ rules, relying on the introduction of $2a-2$ new locations. If the constants in A_S are written in binary, the reduction can be implement in PSPACE. If they are given in unary, the reduction is LOGSPACE.

II. Multiset vector addition systems with states. We write $\vec{a}$, $\vec{b}$, ..., for elements of $\mathbb{N}^m$ or $\mathbb{Z}^m$, called *vectors*. For a vector $\vec{a}$, we write a_1, ..., a_m for its components, that is, $\vec{a} = (a_1, \cdots, a_m)$. We compare vectors componentwise, that is, $\vec{a} \leq \vec{b}$ if and only if $a_1 \leq b_1$, ..., and $a_m \leq b_m$.

A *multiset vector addition system with states* (*MVASS*) is an extension of Petri nets defined as follows : it is a tuple $(Q, m, \delta_1, \delta_2)$ where Q is a finite set of so-called states, $m \in \mathbb{N}$ is the *dimension* of the MVASS, δ_1 is a finite subset of $Q \times \mathbb{Z}^m \times Q$ whose elements are called the *unary transitions*, and δ_2 is a finite subset of $Q \times Q \times Q$ whose elements are called the *binary transitions.*

An MVASS is a compact representation of an infinite transition system, whose set of configurations is $(Q \times \mathbb{N}^m)^{\circledast}$, where $\Sigma^{\circledast}$ denotes the collection of all (finite) multisets over the alphabet Σ. Its transition relation $\to$ is defined by the rules :

$$\begin{array}{rcl} C \uplus \{\!|(q, \vec{a})|\!\} \to C \uplus \{\!|(q', \vec{a} + \vec{d})|\!\} & \qquad & (q, \vec{d}, q') \in \delta_1, \vec{a} + \vec{d} \in \mathbb{N}^m \\ C \uplus \{\!|(q_1, \vec{a}_1), (q_2, \vec{a}_2)|\!\} \to C \uplus \{\!|(q_3, \vec{a}_1 + \vec{a}_2)|\!\} & & (q_1, q_2, q_3) \in \delta_2. \end{array}$$

We will say that $(q, \vec{a})$ is the *active pair* in the first rule, and that the active pairs are $(q_1, \vec{a}_1)$ and $(q_2, \vec{a}_2)$ in the second rule. We also say that the rules are *applied* to its active pairs.

On a computer, one would implement multisets by finite lists, typically.

We order configurations by $\preceq^{\circledast}$, where $(q, \vec{a}) \preceq (q', \vec{a}')$ if and only if $q = q'$ and $\vec{a} \leq \vec{a}'$.

5. Why is $\preceq^{\circledast}$ a wqo ?

Solution:

By the results of the course : $=$ is wqo on Q because Q is finite, $\leq$ is wqo on $\mathbb{N}^k$ by Dickson's Lemma, their product is wqo by Dickson's Lemma (finite products of wqos are wqo), and the $\_^{\circledast}$ construction preserves wqos (that was shown as a consequence of Higman's lemma in the course).

6. Show that every MVASS is a WSTS. *(You must state explicitly what you have to prove. A completely formal argument is not needed, but your proof must be convincing, list all the possible cases that must be examined, and give an idea why what you claim holds in each case.)*

Solution:

We already know that $\preceq^{\circledast}$ is a wqo. We must check monotonicity. This boils down to checking that if $C_1 \to C_2$, then :

- if we add one pair $(q, \vec{a})$ to C_1, then $C_1 \uplus \{\!|(q, \vec{a})|\!\} \to C_2 \uplus \{\!|(q, \vec{a})|\!\}$, and the latter is larger than or equal to C_2 ;
- if we replace $C_1 \stackrel{\text{def}}{=} C \uplus \{(q, \vec{a})\}$ by $C_1' \stackrel{\text{def}}{=} C \uplus \{(q, \vec{a}')\}$ with $(q, \vec{a}) \preceq (q, \vec{a}')$, then $C_1' \to C_2'$ for some C_2' such that $C_2 \preceq^{\circledast} C_2'$, obtained by applying the same rule. There are several cases to consider :
  - The rule map was applied inside C, yielding $C \to C'$: then $C_1' \to C_2'$ where $C_2' \stackrel{\text{def}}{=} C' \uplus \{(q, \vec{a}')\}$; since $C_2 = C' \uplus \{(q, \vec{a})\}$, we have $C_2 \preceq^{\circledast} C_2'$.
  - Or the rule was unary and applied to $(q, \vec{a})$: then $C_2 = C \uplus \{(q', \vec{a} + \vec{d})\}$, where $(q, \vec{d}, q') \in \delta_1$ and $\vec{a} + \vec{d} \geq 0$. We define C_2' as $C \uplus \{(q', \vec{a}' + \vec{d})\}$: then $C_1' \to C_2'$, and $C_2 \preceq^{\circledast} C_2'$.
  - Or the rule was binary and one of its active pairs was $(q, \vec{a})$. In other words, $C = C_0 \uplus \{(q_1, \vec{a}_1)\}$, $C_1 = C_0 \uplus \{(q_1, \vec{a}_1), (q, \vec{a})\}$, $C_2 = C_0 \uplus \{(q_3, \vec{a}_1 + \vec{a})\}$ where $(q_1, q, q_3) \in \delta_2$ (or $(q, q_1, q_3) \in \delta_2$). We define C_2' as $C_0 \uplus \{(q_3, \vec{a}_1 + \vec{a}')\}$.

7. Using WSTS theory, show that the termination problem for MVASS is decidable. *(As for question 3, you must list the properties of the ordering you rely on and justify them.)*

Solution:

We use Proposition 1.36 of the lecture notes. We have just shown that every MVASS is a WSTS. Imagine-finiteness and post-effectiveness are obvious, and the ordering $\preceq^{\circledast}$ is decidable : for example, in order to decide whether $C \preceq^{\circledast} C'$, we look for an injective mapping f from the indices of the first multiset C, represented as a list, to those of the second one C', such that the element $(q_i, \vec{a}_i)$ at every position i of the first list is less than or equal to the element at position $f(i)$ of the second list.

8. Recall what the coverability problem is on MVASS, and show that it is decidable. *(For this, recall precisely what is the statement of the theorem you use. Some assumptions must be verified in order to apply this theorem. For effectiveness assumptions you have to provide an algorithm : that algorithm must be explicit, and must be correct. You need not give a complete formal correctness proof but must provide the intuition of why the algorithm is correct.)*

Solution:

INPUT : two configurations C_0, C_∞ ;
QUESTION : does $C_0 \to^* C$ for some configuration C such that $C_\infty \preceq^{\circledast} C$?

We have alread shown that $\preceq^{\circledast}$ is decidable. We should now verify that the MVASS has effective pred-basis.

Let C' be any configuration. We need to find a finite basis of the set of configurations C such that $C \to D \in \uparrow C'$ for some D. We compute this as the union of :

- for each unary transition $(q, \vec{d}, q') \in \delta_1$, a finite basis of the set of configurations $C \cup \{(q, \vec{a})\}$ such that $D \preceq^{\circledast} C \cup \{(q', \vec{a} + \vec{d})\}$ is larger than or equal to C'. This finite basis consists of all configurations obtained from C' by replacing a pair $(q', \vec{a}')$ in it (with the same q') by $(q, \vec{a})$ where $a_i \stackrel{\text{def}}{=} \max(a'_i - d_i)$ for each i, *plus* $C' \uplus \{\!|(q, \vec{a})|\!\}$ where $\vec{a}$ is the least element of $\mathbb{N}^k$ such that $\vec{a} + \vec{d} \geq 0$ (namely $a_i \stackrel{\text{def}}{=} \max(-d_i, 0)$; to see that this is needed, look at the case where C' is empty) ;
- for each binary transition $(q_1, q_2, q_3) \in \delta_2$, we compute the set of all configurations obtained by replacing a pair $(q_3, \vec{a}')$ (with the given q_3) by two pairs $(q_1, \vec{a}_1)$ and $(q_2, \vec{a}_2)$, where $\vec{a}_1 + \vec{a}_2 = \vec{a}'$ (there are only finitely many choices for $\vec{a}_1$ and $\vec{a}_2$, since both must be below $\vec{a}'$), *plus* the configuration $C' \uplus \{\!|(q_1, \vec{0}), (q_2, \vec{0})\{\!|$ (again, to see that it is needed, consider the case where C' is empty).

Then Proposition 1.38 of the lecture notes tells us that MVASS coverability is decidable.

III. Lengths of r-bad sequences. We consider a generalisation of good and bad sequences. For $r \in \mathbb{N}$, we say that a sequence $a_0, a_1, \ldots$ over a qo $(A, \leq)$ is r-good if it contains an increasing subsequence of length $r + 1$, i.e. if there exist $r + 1$ indices $i_0 < \cdots < i_r$ s.t. $a_{i_0} \leq \cdots \leq a_{i_r}$. A sequence is r-bad if it is not r-good. Thus “1-good” and “1-bad” correspond to the usual notions of “good” and “bad” for sequences over a qo.

Assume A is a normed wqo. We define $L_{g,r,A}(n)$ as the maximal length of a (g, n)-controlled r-bad sequence over A, generalizing the notation $L_{g,A}(n)$ seen in class.

9. Explain briefly why $L_{g,r,A}(n)$ is a well-defined natural number.

Solution:

The tree of controlled r-bad sequences is finitely branching (by virtue of control) and all its branches are finite (since r-bad sequences are finite in a wqo). Hence it is a finite tree and has a well-defined maximal length for its branches.

10. Consider a (g, n)-controlled r-bad sequence $s = a_0, a_1, \ldots, a_\ell$ of maximal length. Show that it is r'-good for any $r' < r$.

Solution:

Assume $r > 0$ and let p be the maximal value s.t. s is p-good ; thus $p < r-1$. If we define s' by extending s with $r-p-1$ copies of a_0 we obtain a sequence that is (g,n)-controlled and r-bad. (Indeed, if s' is r-good and contains an increasing subsequence of length r, by removing the last $r-p-1$ elements of the subsequence we see that s was $(p+1)$-good.) Now, if $r-p-1 > 0$, s' is longer than s, contradicting the maximality of s. We conclude that $p = r-1$.

Recall that, for $k \in \mathbb{N}$, Γ_k is the nwqo whose elements are the letters $\{b_0, b_1, .., b_{k-1}\}$, of zero norm, with trivial ordering Id_{Γ_k}. We want to reduce $L_{g,r,A}$ to $L_{g,A\times\Gamma_r}$.

11. Show that $L_{g,r,A}(n) \leq L_{g,A\times\Gamma_r}(n)$.

Solution:

Pick an r-bad sequence $s = a_0, a_1, \ldots, a_\ell$ over A of maximal length. Say that an index i in $0, \ldots, \ell$ is *p-good* (in s) if it can start an increasing subsequence of length $p+1$, i.e. if there exist indices $i = i_0 < \cdots < i_p$ s.t. $a_{i_0} \leq \cdots \leq a_{i_p}$. The *goodness* $\gamma(i)$ of an index i is the largest p s.t. i is p-good in s.

Since s is r-bad, all the indices are at most $(r-1)$-good, hence $\gamma(i) < r$ for all i. Note also that $i < j$ and $a_i \leq a_j$ imply $\gamma(i) > \gamma(j)$ since any good subsequence starting at index j can be made longer by starting with a_i at index i.

We now define a sequence s' over $A \times \Gamma_r$:

$$s' = \langle a_0, b_{\gamma(0)}\rangle, \langle a_1, b_{\gamma(1)}\rangle, \ldots, \langle a_\ell, b_{\gamma(\ell)}\rangle \ .$$

s' is $g(n)$-controlled since s is, and since $|\langle a, b_i\rangle|_{A\times\Gamma_r} = \max(|a|_A, |b_i|_{\Gamma_r}) = \max(|a|_A, 0) = |a|_A$. (Once more we see that the norm of letters and the norm of pairs was well chosen.)

Furthermore s' is bad since, as noted $\gamma(i) \neq \gamma(j)$ when $i < j$. Thus s' is a witness showing $L_{g,r,A}(n) \leq L_{g,A\times\Gamma_r}(n)$.

12. Show that $L_{g,r,A}(n) \geq L_{g,A\times\Gamma_r}(n)$.

Solution:

Take $s = \langle a_0, b_0\rangle, \ldots, \langle a_\ell, b_\ell\rangle$ a (g,n)-controlled bad sequence of maximal length over $A\times\Gamma_r$ and project over A to obtain the sequence $s' = a_0, \ldots, a_\ell$.

We claim that s' is (g,n)-controlled and r-bad. The condition on the control is immediate since s is controlled and $|a_i|_A = |\langle a_i, b_i\rangle|_{A\times\Gamma_r}$. Regarding badness, assume s' to be r-good : then there exist $r+1$ indices $0 \leq i_0 < \cdots < i_r \leq \ell$ s.t. $a_{i_0} \leq \cdots \leq a_{i_r}$. By the pigeonhole principle, there exists some $b \in \Gamma_r$ that appears at least twice among $\{b_{i_0}, \ldots, b_{i_r}\}$, i.e. there exist two indices $i_j < i_k$ in $\{i_0, \ldots, i_r\}$ s.t. $b_{i_j} = b_{i_k}$. Thus $\langle a_{i_j}, b_{i_j}\rangle \leq_{A\times\Gamma_r} \langle a_{i_k}, b_{i_k}\rangle$, contradicting the badness of s.

13. Write more simply $L(n)$ for $L_{g,A}(n)$, and n' for $g^{L(n)}(n)$.
Does $L_{g,2,A}(n) \leq L(n) + L(n')$ hold for all A, g, n ?
And does $L_{g,2,A}(n) \geq L(n) + L(n')$?

Solution:

The second inequality holds : if $s = a_0, \ldots, a_\ell$ is bad and (g,n)-controlled, if $s' = b_0, \ldots, b_m$ is bad and (g,n') controlled for $n' = L(g^{\ell+1}(n))$, then $s \cdot s'$ is (g,n)-controlled and 2-bad.

The first equality does not always hold. E.g., if we take $g = \mathit{Succ}$ and $A = \{a, b\}$ with trivial ordering Id_A, and with norm given by $|a| = 0$ and $|b| = 2$, the sequence a, a, b, b is 2-bad and $(g,0)$-controlled, hence $L_{g,2,A}(0) \geq 4$. However $L(0) = 1$ since a, b is not $(g,0)$-controlled, and $L(1) = 2$. For $n = 0$ this gives $L(n) + L(g^{L(n)}(n)) = L(0) + L(1) = 1+2 = 3 < L_{g,2,A}(n)$.