Preorder-Constrained Simulation for Nondeterministic Automata

Koko Muroya ✉
RIMS, Kyoto University, Japan

Takahiro Sanada ✉
RIMS, Kyoto University, Japan

Natsuki Urabe ✉
National Institute of Informatics, Tokyo, Japan

Abstract

We describe our ongoing work on generalizing some quantitatively constrained notions of weak simulation up-to that are recently introduced for deterministic systems modeling program execution. We present and discuss a new notion dubbed *preorder-constrained simulation* that allows comparison between words using a preorder, instead of equality.

2012 ACM Subject Classification Theory of computation → Verification by model checking

Keywords and phrases simulation, weak simulation, up-to technique, language inclusion, preorder

Digital Object Identifier 10.4230/LIPIcs.CALCO.2021.21

Category Early Ideas

Funding The first and second authors are supported by JST, ACT-X Grant No. JPMJAX190U, Japan. The third author is supported by JST ERATO HASUO Metamathematics for Systems Design Project (No. JPMJER1603).

1 Introduction: Simulation Notions with Bounded Number of Steps

In the literature of program semantics, coinductive techniques have often been used to establish equivalence between program behaviors. A recent approach utilizes weak simulations with quantitative constraints on the length of terminating runs. These constraints enable comparison of execution cost for programs, in terms of the number of execution steps it takes for a program to terminate.

One example is Accattoli et al.'s notion called *improvement* [1]. It was used to show that certain rewriting of a program before execution not only preserves the execution result, but also *improves* the execution cost by requiring less execution steps. Another example was used in the first author's previous work [9]. It is dubbed (Q, Q_1, Q_2)*-simulation*, parameterized by a triple (Q, Q_1, Q_2) of preorders on natural numbers, i.e. on lengths of runs. The first preorder Q is used to compare lengths of accepted runs, and it generalizes the "greater-than-or-equal" preorder $\geq$ used by improvements. The other two preorders Q_1, Q_2 are for additionally incorporating the so-called *up-to* technique. Subtle conditions on these preorders are identified in *loc. cit.* to make the combination of weak simulations and the up-to technique work.

These two notions are both designed for unlabeled deterministic transition systems, which can model execution of deterministic programs only. We aim to pursue the idea of constraining terminating, or accepted, runs, in a more general setting. This abstract describes our ongoing work on generalizing (Q, Q_1, Q_2)-simulations to nondeterministic automata. We present a novel notion of *preorder-constrained simulation* that is a weak simulation up-to constrained by preorders on words, not on natural numbers. It entails a generalized notion of language inclusion that compares words using a preorder instead of equality.

9th Conference on Algebra and Coalgebra in Computer Science (CALCO 2021).
Editors: Fabio Gadducci and Alexandra Silva; Article No. 21; pp. 21:1–21:5
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

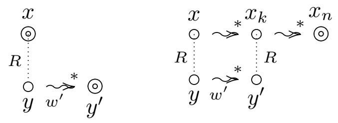

Figure 1 Def. 3 (**Final** and **Step**).

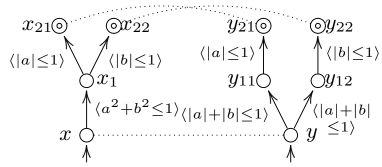

Figure 2 Ex. 4.

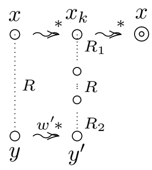

Figure 3 Def. 9.

2 Main Contribution

Let $A_k = (X_k, \Sigma, \rightsquigarrow_k \subseteq X_k \times \Sigma \times X_k, F_k \subseteq X_k)$ $(k \in \{1,2\})$ be nondeterministic automata, $x \in X_1$ and $y \in X_2$, and $L^*_{A_1}(x), L^*_{A_2}(y) \subseteq \Sigma^*$ be the set of words accepted from x and y respectively. The ordinary simulation notion [7] proves language inclusion $L^*_{A_1}(x) \subseteq L^*_{A_2}(y)$. Instead, our simulation notion proves $\mathcal{Q}$*-trace inclusion*.

▶ **Def. 1.** For a preorder $\mathcal{Q} \subseteq \Sigma^* \times \Sigma^*$, we write $x \preceq_{\mathcal{Q}} y$ and say $\mathcal{Q}$-trace inclusion holds between x and y when $\forall w \in L^*_{A_1}(x).\ \exists w' \in L^*_{A_2}(y).\ w\mathcal{Q}w'$.

▶ **Example 2.**

i) when $\mathcal{Q}$ is the equality, $x \preceq_{\mathcal{Q}} y$ iff $L^*_{A_1}(x) \subseteq L^*_{A_2}(y)$.

ii) When Σ contains a special letter τ, and $w\mathcal{Q}w'$ means that w and w' are the same except for τ, then $x \preceq_{\mathcal{Q}} y$ iff *weak language inclusion*, i.e. language inclusion ignoring τ, holds.

iii) When $w\mathcal{Q}w'$ means that w is a subword of w', $x \preceq_{\mathcal{Q}} y$ iff for each $w \in L^*_{A_1}(x)$ there exists $w' \in L^*_{A_2}(y)$ such that w is a subword of w'.

iv) When Σ is the powerset 2^{AP} of some set AP and $a_1 \ldots a_k \mathcal{Q} a'_1 \ldots a'_{k'}$ means that $k = k'$ and $a_i \subseteq a'_i$ for each $i \in \{1, \ldots, k\}$, then $x \preceq_{\mathcal{Q}} y$ iff for each $a_1 \ldots a_k \in L^*_{A_1}(x)$ there exists $a'_1 \ldots a'_k \in L^*_{A_2}(y)$ such that $a_i \subseteq a'_i$ for each $i \in \{1, \ldots, k\}$.

2.1 Preorder-Constrained Simulation without up-to

We hereby introduce a new simulation notion for witnessing $\mathcal{Q}$-trace inclusion.

▶ **Def. 3.** We call $R \subseteq X_1 \times X_2$ a $\mathcal{Q}$*-constrained simulation* from A_1 to A_2 if, for any $(x, y) \in R$, the following holds (see also Fig. 1).

Final: If $x \in F_1$ then there exist $w' \in \Sigma^*$ and $y' \in F_2$ such that $\varepsilon\mathcal{Q}w'$ and $y \overset{w}{\rightsquigarrow}{}^*_2\, y'$.

Step: For each $a_1 \ldots a_n \in \Sigma^+$ and $x_1 \ldots x_n \in X_1^+$ such that $x \overset{a_1}{\rightsquigarrow} x_1 \overset{a_2}{\rightsquigarrow} \cdots \overset{a_n}{\rightsquigarrow} x_n$ and $x_n \in F_1$, there exist $k \in \{1, \ldots, n\}$, $w' \in \Sigma^*$ and $y' \in X_2$ such that $a_1, \ldots, a_k \mathcal{Q} w'$, $y \overset{w'}{\rightsquigarrow}{}^*_2\, y'$ and i) $x_k R y'$ or ii) $k = n$ and $y' \in F_2$.

▶ **Example 4.** We continue Ex. 2(iv). Let $\mathsf{AP} := \mathbb{R}^2$. For a formula $\varphi(a, b)$ with free variables a and b, let $\langle\varphi\rangle := \{(a, b) \in \mathbb{R}^2 \mid \varphi(a, b)\}$. For nondeterministic automata illustrated in Fig. 2, as $\langle a^2 + b^2 \le 1\rangle \subseteq \langle |a| + |b| \le 1\rangle$, $R := \{(x, y), (x_{21}, y_{21}), (x_{22}, y_{22})\}$ is a $\mathcal{Q}$-constrained simulation. Note that x_1, y_{11} and y_{12} are not involved by R.

▶ **Prop. 5** (soundness)**.** *If $\mathcal{Q}$ is closed under concatenation (i.e. $w_1\mathcal{Q}w'_1$ and $w'_2\mathcal{Q}w'_2$ imply $w_1w_2\mathcal{Q}w'_1w'_2$), xRy implies $x \preceq_{\mathcal{Q}} y$.* ◀

Unfortunately, it seems that $\mathcal{Q}$-constrained simulation is not practicable as it is hard to check if given R satisfies **Step** in Def. 3 for all $a_1 \ldots a_n \in \Sigma^+$ and $x_1 \ldots x_n \in X^+$. In fact, by letting $R := \{(x, y) \mid x \preceq_{\mathcal{Q}} y\}$, we can easily see that $\preceq_{\mathcal{Q}}$ is a $\mathcal{Q}$-constrained simulation.

▶ **Prop. 6** (completeness)**.** *If $\mathcal{Q}$ is closed under concatenation, there exists a $\mathcal{Q}$-constrained simulation R such that $x \preceq_{\mathcal{Q}} y$ implies xRy.* ◀

Position	Player	Moves	
$(w, x, y) \in \Sigma^* \times X_1 \times X_2$	Challenger	(wa, x', y) s.t. $x \overset{a}{\rightsquigarrow}_1 x'$	choose a successor state and enqueue the label
		$(\checkmark, w, x, y)$ when $x \in F_1$	declare the last turn
$(w, x', y) \in \Sigma^+ \times X_1 \times X_2$	Simulator	(w, x', y)	skip the turn
		(ε, x', y') s.t. $\exists w' \in \Sigma^*.\ y \overset{w'}{\rightsquigarrow}{}^*_2\ y'$ and $w \mathcal{Q} w'$	dequeue all, and simulate it
$(\checkmark, w, x, y) \in \{\checkmark\} \times \Sigma^+ \times X_1 \times X_2$		sim-win if $\exists w' \in \Sigma^*.\ y \overset{w'}{\rightsquigarrow}{}^*_2\ y'$, $w \mathcal{Q} w'$ and $y' \in F_2$	dequeue all and simulate it so that accepting state is reached

Figure 4 Two-player game characterizing $\mathcal{Q}$-constrained simulation. Simulator wins if sim-win is reached, Challenger gets stuck, or a play continues infinitely.

This means that existence of a $\mathcal{Q}$-constrained simulation relating two states is very difficult to determine in many cases. For example, ordinary language inclusion between nondeterministic automata (i.e. $\mathcal{Q}$-trace inclusion when $\mathcal{Q}$ is the equality) is known to be PSPACE-complete [8]. We therefore consider approximating $\mathcal{Q}$-constrained simulation. We consider fixing $M \in \mathbb{N}$ and replacing **Step** with the following (here $A^{[m,M]} := \bigcup_{m \le i \le M} A^i$):

Step$^{\le M}$**:** For each $a_1 \dots a_n \in \Sigma^{[1,M]}$ and $x_1 \dots x_n \in X_1^{[1,M]}$ such that $x \overset{a_1}{\rightsquigarrow} x_1 \overset{a_2}{\rightsquigarrow} \cdots \overset{a_n}{\rightsquigarrow} x_n$ and either $x_n \in F_1$ or $n = M$, there exist $k \in \{1, \dots, n\}$, $w' \in \Sigma^*$ and $y' \in X_2$ such that $a_1, \dots, a_k \mathcal{Q} w'$, $y \overset{w'}{\rightsquigarrow}{}^*_2\ y'$ and i) $x_k R y'$ or ii) $k = n < M$ and $y' \in F_2$.

▶ **Prop. 7.** *Given $x \in X_1$ and $y \in X_2$, existence of R satisfying **Final**, **Step**$^{\le M}$ and xRy can be checked in polynomial time to $|\Sigma|$, $|X_1|$, $|X_2|$ and $\mathcal{T}_M(|\Sigma|, |X_2|)$. Here $\mathcal{T}_M(p, q)$ is the computation time for the following problem: given $w \in \Sigma^{[1,M]}$ and a nondeterministic automaton whose alphabet and state space have sizes of p and q, check if the set $\{w' \mid w \mathcal{Q} w'\}$ intersects with the language of the automaton.* ◀

As **Step**$^{\le M}$ implies **Step**, Prop. 5 still holds after the modification. Moreover, by Prop. 7, if M is fixed and $\mathcal{T}_M(|\Sigma|, |X_2|)$ is polynomial to $|X_2|$ and $|\Sigma|$ (it holds for all $\mathcal{Q}$ illustrated above), then existence of a $\mathcal{Q}$-constrained simulation relating two states can be checked in polynomial time.

We conclude this section by giving a game theoretic characterization for $\mathcal{Q}$-constrained simulations, namely the safety game in Fig. 4 played by Challenger and Simulator.

▶ **Prop. 8.** *Simulator is winning in the two-player game in Fig. 4 from a state (ε, x, y) if and only if there exists a $\mathcal{Q}$-constrained simulation R such that xRy.* ◀

Intuitively, $w \in \Sigma^*$ in Fig. 4 is understood as a *queue* that saves labels executed on A_1 by Challenger. Basically, Simulator can skip the turn until she can construct a path labeled by a word $w' \in \Sigma^*$ such that $w \mathcal{Q} w'$. However, when an accepting state is reached on A_1, Challenger can also declare the last turn and force Simulator to construct a path immediately, although if Simulator succeeded in constructing a path then Challenger loses.

The modification of $\mathcal{Q}$-constrained simulation stated above, i.e. replacing **Step** with **Step**$^{\le M}$, corresponds to replacing Σ^* and Σ^+ with $\Sigma^{[0,M]}$ and $\Sigma^{[1,M]}$ respectively, and prohibiting Simulator from skipping the turn when $|w| = M$ in Fig. 4.

2.2 Preorder-Constrained Simulation with up-to

We can think of an up-to variant of $\mathcal{Q}$-constrained simulations from A_1 to A_2.

▶ **Def. 9.** Let $R_1 \subseteq X_1 \times X_1$ and $R_2 \subseteq X_2 \times X_2$. A $\mathcal{Q}$-constrained simulation R *up-to* (R_1, R_2) is defined in almost the same manner as Def. 3, except that $x_k R y'$ at the end of **Step** is replaced by $x_k R_1 R R_2 y'$ (see also Fig. 3).

We are in particular interested in $R_1 \subseteq \preceq_{\mathcal{Q}_1}$ and $R_2 \subseteq \preceq_{\mathcal{Q}_2}$. Naturally, R_1 and R_2 cannot be arbitrary to verify soundness $xRy \implies x \preceq_{\mathcal{Q}} y$. They have to be *compatible* with $\preceq_{\mathcal{Q}}$. We should also be aware of that a naïve combination of weak simulations and up-to techniques is known to be unsound, and requires special care [10, 11]. Ex. 2(ii) suggests that $\mathcal{Q}$-constrained simulation is a variant of weak simulation. It turns out that restrictions can be simply on the preorders $\mathcal{Q}, \mathcal{Q}_1, \mathcal{Q}_2$.

▶ **Prop. 10.** *Assume $\mathcal{Q}$ is closed under concatenation. If $R_1 \subseteq \preceq_{\mathcal{Q}_1}$ and $R_2 \subseteq \preceq_{\mathcal{Q}_2}$ for preorders $\mathcal{Q}_1, \mathcal{Q}_2 \subseteq \Sigma^* \times \Sigma^*$ satisfying the following conditions, then xRy implies $x \preceq_{\mathcal{Q}} y$: i) $\mathcal{Q}_1\mathcal{Q}\mathcal{Q}_2 \subseteq \mathcal{Q}$; and ii) $w\mathcal{Q}_1 w'$ implies $|w| \geq |w'|$.* ◀

Cond. (i) ensures compatibility of $\preceq_{\mathcal{Q}_1}, \preceq_{\mathcal{Q}_2}$ with $\preceq_{\mathcal{Q}}$, and Cond. (ii) ensures safe integration of the up-to technique. They are strongly inspired by (Q, Q_1, Q_2)-simulation for unlabeled and deterministic automata [9].

3 Related Work

The above notion is similar to *buffered simulation* [3], which was developed to enable more relations to witness language inclusion. Buffered simulations allow Simulator to skip his turn, to buffer Challenger's moves and to simulate them later together, which has a similar flavor to our simulation notion (cf. Ex. 4). Hence our simulation notion can be also thought of as a generalization of buffered simulation.

Preorder-constrained simulations allow a quantitative reasoning such as comparing lengths of accepted runs. There exist quantitative simulation notions for comparing costs of weighted automata. Many of them are for probabilistic systems [6, 5, 4]. One simulation notion for automata weighted with costs was introduced as a matrix over real numbers [12]. A methodology for comparing infinite runs of weighted automata is also known [2]. In contrast to weighted automata, which are labeled with both letters and weights, our target is automata labeled with letters only. Quantities appear in the set of words, in our approach.

4 Research Directions

Our simulation notion focuses on finite languages. As is the case for the ordinary simulation notion, our notion may fail to prove inclusion of finite languages when there is no inclusion of infinite languages. We are looking into possible solutions.

We suspect that Cond. (ii) of Prop. 5, whose analogues are also in existing notions of weak simulation up-to, is too strong. We think $\mathcal{Q}_1$ violating Cond. (ii) can be allowed finitely many times. However, at the same time, we should note that the relaxation makes the definition of simulations a global one, which can result in a more complicated algorithm for finding it. We should make sure that it does not ruin efficiency gained by up-to techniques.

Ex. 4 suggests that our simulation notion works well with systems whose alphabet Σ carries an order. Such a system also arises in the study of linear temporal logic (LTL). An LTL formula induces a Büchi automaton labeled with the powerset 2^{AP} of atomic propositions [13]. The alphabet 2^{AP} is ordered by the inclusion, which induces a preorder on $(2^{\mathsf{AP}})^*$.

We are also interested in a categorical study of our simulation notion. One possible strategy would be to use the category **PreOrd** of preordered sets as the base category. The nondeterministic branching would be then captured by the powerset functor (or possibly a monad) $\mathcal{P}$ lifted to **PreOrd**. The categorical generalization might allow us to transfer our simulation notion to systems with other branching types, e.g. probabilistic one.

References

1 Beniamino Accattoli, Ugo Dal Lago, and Gabriele Vanoni. The machinery of interaction. In *PPDP 2020*, pages 4:1–4:15. ACM, 2020.

2 Suguman Bansal, Swarat Chaudhuri, and Moshe Y. Vardi. Comparator automata in quantitative verification. In *FoSSaCS 2018*, volume 10803 of *Lecture Notes in Computer Science*, pages 420–437. Springer, 2018.

3 Milka Hutagalung, Martin Lange, and Étienne Lozes. Buffered simulation games for büchi automata. In Zoltán Ésik and Zoltán Fülöp, editors, *AFL 2014*, volume 151 of *EPTCS*, pages 286–300, 2014.

4 Bart Jacobs and Jesse Hughes. Simulations in coalgebra. *Electronic Notes in Theoretical Computer Science*, 82(1):128–149, 2003.

5 Bengt Jonsson and Kim Guldstrand Larsen. Specification and refinement of probabilistic processes. In *LICS 1991*, pages 266–277. IEEE Computer Society, 1991.

6 Kim G. Larsen and Arne Skou. Bisimulation through probabilistic testing. *Information and Computation*, 94(1):1–28, 1991.

7 Nancy A. Lynch and Frits W. Vaandrager. Forward and backward simulations: I. Untimed systems. *Inf. Comput.*, 121(2):214–233, 1995. `doi:10.1006/inco.1995.1134`.

8 A. R. Meyer and L. J. Stockmeyer. The equivalence problem for regular expressions with squaring requires exponential space. In *13th Annual Symposium on Switching and Automata Theory (swat 1972)*, pages 125–129, 1972. `doi:10.1109/SWAT.1972.29`.

9 Koko Muroya. *Hypernet Semantics of Programming Languages.* PhD thesis, University of Birmingham, 2020.

10 Damien Pous. Up-to techniques for weak bisimulation. In Luís Caires, Giuseppe F. Italiano, Luís Monteiro, Catuscia Palamidessi, and Moti Yung, editors, *ICALP 2005*, volume 3580 of *Lecture Notes in Computer Science*, pages 730–741. Springer, 2005.

11 Damien Pous. New up-to techniques for weak bisimulation. *Theoretical Computer Science*, 380(1):164–180, 2007. Automata, Languages and Programming.

12 Natsuki Urabe and Ichiro Hasuo. Generic forward and backward simulations III: quantitative simulations by matrices. In *CONCUR 2014*, volume 8704 of *Lecture Notes in Computer Science*, pages 451–466. Springer, 2014.

13 Moshe Y. Vardi and Pierre Wolper. Reasoning about infinite computations. *Information and Computation*, 115(1):1–37, 1994.