

# 1 The Effect of the Acceptance Density

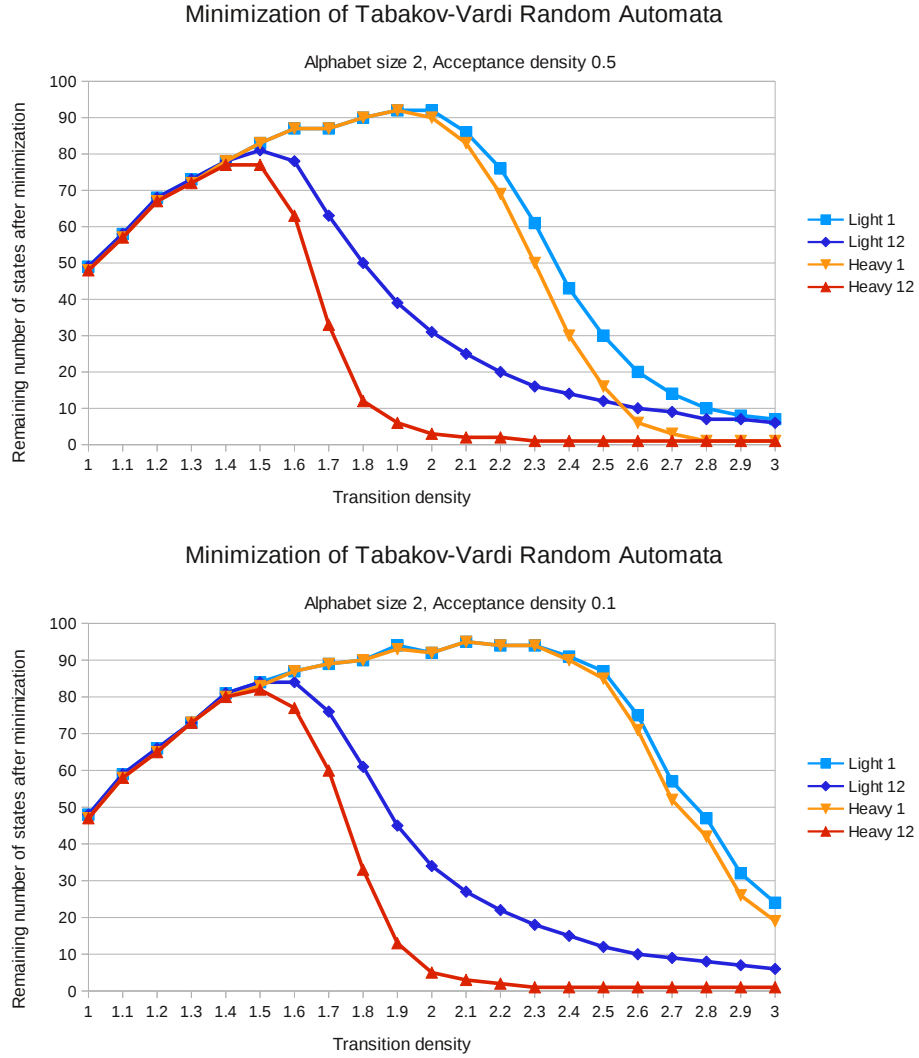


Figure 1: Minimization of Tabakov-Vardi random automata with  $n = 100$ ,  $|\Sigma| = 2$ ,  $ad = 0.5$  (top),  $ad = 0.1$  (bottom) and varying  $td$ . We use the Light 1, Light 12, Heavy 1 and Heavy 12 methods and plot the average number of states of the minimized automata. Every point in the top (resp. bottom) graph the average of 1000 (resp. 300) automata. Note how a small acceptance density makes minimization harder without lookahead, but not much harder for lookahead 12.

## 2 Scalability

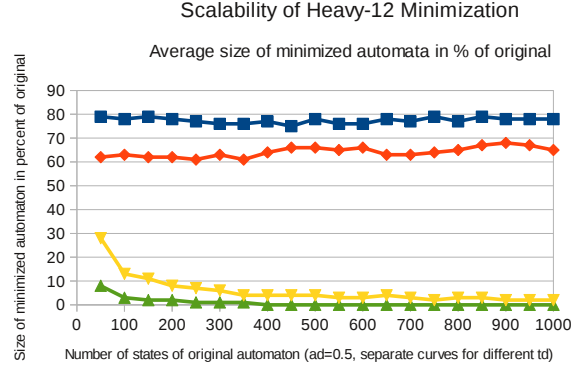


Figure 2: Minimization of Tabakov-Vardi random automata with  $ad = 0.5$ ,  $|\Sigma| = 2$ , and increasing  $n = 50, 100, \dots, 1000$ . Different curves for different  $td$ . We plot the average size of the Heavy-12 minimized automata, in percent of their original size. Every point is the average of 300 automata. Note that the lookahead of 12 does not change, i.e., larger automata do not require a higher lookahead for a good minimization.

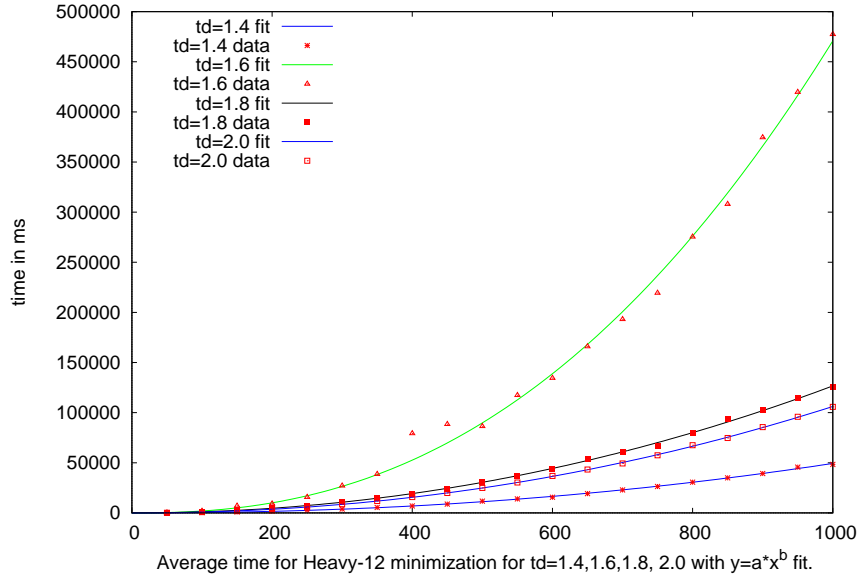


Figure 3: Minimization of Tabakov-Vardi random automata as above. Here we plot the average computation time (in ms) for the minimization, and a least-squares fit by the function  $a * n^b$ . For  $td = 1.4, 1.6, 1.8, 2.0$  we obtain  $0.018 * n^{2.14}$ ,  $0.32 * n^{2.39}$ ,  $0.087 * n^{2.05}$  and  $0.055 * n^{2.09}$ , respectively.

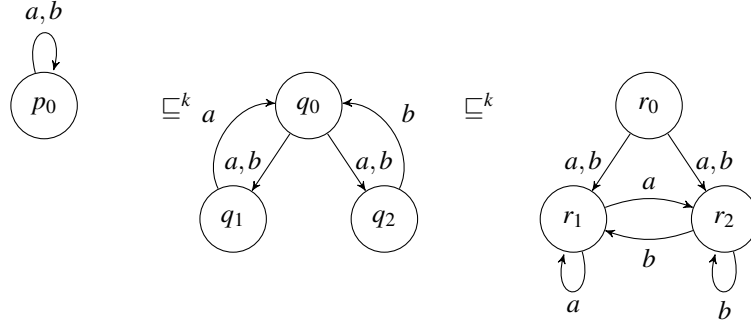


Figure 4: Lookahead simulation is not transitive.

### 3 Non-transitivity of lookahead simulation

Lookahead simulation for  $k \geq 2$  is not transitive. Consider the example in Figure 4. We have  $p_0 \sqsubseteq^k q_0 \sqsubseteq^k r_0$  (and  $k = 2$  suffices), but  $p_0 \not\sqsubseteq^k r_0$  for any  $k > 0$ :

- $p_0 \sqsubseteq^k q_0$ , with  $k = 2$ : Duplicator takes the transition via  $q_1$  or  $q_2$  depending on whether Spoiler plays word  $(a + b)a$  or  $(a + b)b$ , respectively.
- $q_0 \sqsubseteq^k r_0$ , with  $k = 2$ : If Spoiler goes to  $q_1$  or  $q_2$ , then Duplicator goes to  $r_1$  or  $r_2$ , respectively. That  $q_1 \sqsubseteq^k r_1$  holds can be shown as follows (the case  $q_2 \sqsubseteq^k r_2$  is similar). If Spoiler takes transitions  $q_1 \xrightarrow{a} q_0 \xrightarrow{a} q_1$ , then Duplicator does  $r_1 \xrightarrow{a} r_1 \xrightarrow{a} r_1$ , and if Spoiler does  $q_1 \xrightarrow{a} q_0 \xrightarrow{b} q_1$ , then Duplicator does  $r_1 \xrightarrow{a} r_2 \xrightarrow{b} r_1$ . The other cases are similar.
- $p_0 \not\sqsubseteq^k r_0$ , for any  $k > 0$ . From  $r_0$ , Duplicator can play a trace for any word  $w$  of length  $k > 0$ , but in order to extend it to a trace of length  $k + 1$  for any  $w' = wa$  or  $wb$ , she needs to know whether the last  $(k + 1)$ -th symbol is  $a$  or  $b$ . Thus, no finite lookahead suffices for Duplicator.

Incidentally, notice that  $r_0$  simulates  $p_0$  with  $k$ -continuous simulation, and  $k = 2$  suffices.

## 4 Fixed-point characterization of lookahead simulation

Lookahead simulation enjoys the following preservation property which allows a fix-point characterization: Let  $x \in \{\text{di}, \text{de}, \text{f}, \text{bw}\}$  and  $k > 0$ . When Duplicator plays according to a winning strategy, in any configuration  $(p_i, q_i)$  of the resulting play,  $p_i \sqsubseteq^{k-x} q_i$ . Thus,  $k$ -lookahead simulation (without acceptance condition) can be characterized as the largest  $X \subseteq Q \times Q$  which is closed under a certain monotone predecessor operator. For convenience, we take the point of view of Spoiler, and compute the complement relation  $W^x = (Q \times Q) \setminus \sqsubseteq^{k-x}$  instead. This is particularly useful for delayed simulation, since we can avoid recording the obligation bit (see [1]) by using the technique of [2].

**Direct and backward simulation.** Consider the following predecessor operator  $\text{CPre}^{\text{di}}(X)$ , for any set  $X \subseteq Q \times Q$ :

$$\begin{aligned} \text{CPre}^{\text{di}}(X) = \{ & (p_0, q_0) \mid \exists (p_0 \xrightarrow{a_0} p_1 \xrightarrow{a_1} \dots \xrightarrow{a_{k-1}} p_k) \\ & \forall (q_0 \xrightarrow{a_0} q_1 \xrightarrow{a_1} \dots \xrightarrow{a_{m-1}} q_m), 0 < m \leq k, \\ & \text{either } \exists (0 \leq j \leq m) \cdot p_j \in F \text{ and } q_j \notin F, \\ & \text{or } (p_m, q_m) \in X \} \end{aligned}$$

Intuitively,  $(p, q) \in \text{CPre}^{\text{di}}(X)$  iff, from position  $(p, q)$ , in one round of the game Spoiler can either force the game in  $X$ , or violate the winning condition for direct simulation. For backward simulation,  $\text{CPre}^{\text{bw}}(X)$  is defined analogously, except that transitions are reversed and also initial states are taken into account:

$$\begin{aligned} \text{CPre}^{\text{bw}}(X) = \{ & (p_0, q_0) \mid \exists (p_0 \xleftarrow{a_0} p_1 \xleftarrow{a_1} \dots \xleftarrow{a_{k-1}} p_k) \\ & \forall (q_0 \xleftarrow{a_0} q_1 \xleftarrow{a_1} \dots \xleftarrow{a_{m-1}} q_m), 0 < m \leq k, \\ & \text{either } \exists (0 \leq j \leq m) \cdot p_j \in F \text{ and } q_j \notin F, \\ & \text{or } \exists (0 \leq j \leq m) \cdot p_j \in I \text{ and } q_j \notin I, \\ & \text{or } (p_m, q_m) \in X \} \end{aligned}$$

**Remark** The definition of  $\text{CPre}^x(X)$  requires that the automaton has no deadlocks; otherwise, Spoiler would incorrectly lose if she can only perform at most  $k' < k$  transitions. We assumed that the automaton is complete to keep the definition simple, but our implementation works with general automata.

For  $X = \emptyset$ ,  $\text{CPre}^x(X)$  is the set of states from which Spoiler wins in at most one step. Thus, Spoiler wins iff she can eventually reach  $\text{CPre}^x(\emptyset)$ . Formally, for  $x \in \{\text{di}, \text{bw}\}$ ,  $W^x = \mu W \cdot \text{CPre}^x(W)$ .

**Delayed and fair simulation.** We introduce a more elaborate three-arguments predecessor operator  $\text{CPre}(X, Y, Z)$ . Intuitively, a configuration belongs to  $\text{CPre}(X, Y, Z)$  iff Spoiler can make a move s.t., for any Duplicator's reply, at least one of the following conditions holds:

1. Spoiler visits an accepting state, while Duplicator never does so; the game goes to  $X$ .
2. Duplicator never visits an accepting state; the game goes to  $Y$ .

3. The game goes to  $Z$ .

$$\begin{aligned}
\text{CPre}(X, Y, Z) = \{ & (p_0, q_0) \mid \exists (p_0 \xrightarrow{a_0} p_1 \xrightarrow{a_1} \dots \xrightarrow{a_{k-1}} p_k) \\
& \forall (q_0 \xrightarrow{a_0} q_1 \xrightarrow{a_1} \dots \xrightarrow{a_{m-1}} q_m) \cdot \forall (0 < m \leq k) \cdot \\
& \text{either } \exists (0 \leq i \leq m) \cdot p_i \in F, \forall (0 \leq j \leq m) \cdot q_j \notin F, (p_m, q_m) \in X \\
& \text{or } \forall (0 \leq j \leq m) \cdot q_j \notin F, (p_m, q_m) \in Y \\
& \text{or } (p_m, q_m) \in Z \}
\end{aligned}$$

For fair simulation, Spoiler wins iff, except for finitely many rounds, she visits accepting states infinitely often while Duplicator does not. Thus,  $W^f = \mu W \cdot \nu X \cdot \mu Y \cdot \text{CPre}(X, Y, W)$ . For delayed simulation, Spoiler wins if, after finitely many rounds, 1) she can visit an accepting state, and 2) she can prevent Duplicator from visiting accepting states in the future. For condition 1), let  $\text{CPre}^1(X, Y) := \text{CPre}(X, Y, Y)$ , and, for 2),  $\text{CPre}^0(X, Y) := \text{CPre}(X, X, Y)$ . Then,  $W^{\text{de}} = \mu W \cdot \text{CPre}^1(\nu X \cdot \text{CPre}^0(X, W), W)$ .

## References

- [1] Kousha Etessami, Thomas Wilke, and Rebecca A. Schuller. Fair Simulation Relations, Parity Games, and State Space Reduction for Büchi Automata. *SIAM J. Comput.*, 34(5):1159–1175, 2005.
- [2] S. Juvekar and N. Piterman. Minimizing Generalized Büchi Automata. In *Computer Aided Verification*, volume 4414 of *LNCS*, pages 45–58. Springer-Verlag, 2006.