Can We Ever Stop Them?

Jörg Desel

FernUniversität in Hagen, Germany
joerg.desel@fernuni-hagen.de

This short article is devoted to all Petri net researchers born in the late 1950s (like myself), which are thus around their respective 60th birthday. In particular, it is devoted to Maciej Koutny.

In industry, many people around 60 are already planning their retirement. The situation in academia is different; most professors at this age seem to be as productive as ever and ignore the benefits of retirement age. So the question tackled in this contribution is not whether it is possible to keep them from work *immediately* but to keep them from work *eventually*. But is there a switch which turns people in a state such that they stop working eventually? In terms of Petri nets, we can formulate this problem as follows:

> Assume a Petri net and a transition t of this net. Given any reachable marking m of the net, can we eventually stop the behavior of the net by forbidding occurrences of t, or, equivalently, does any reachable marking m enable an infinite occurrence sequence without occurrences of t?

Apparently, this question is also highly relevant for real applications of Petri nets. For example, given a robot (or any kind of machine) modeled by a Petri net, can some component modeled by a particular transition be used as a cut out? As known from our computers, immediate stops are not always desirable, but rather forced shut down processes.

The problem tackled in this article could be solved by any standard mechanism involving temporal logics. There exist standard model checking procedures for Petri nets and properties stated as a temporal logic formula. Instead, this article provides a solution which is purely based on Petri net analysis techniques.

Throughout this paper we consider place/transition Petri nets without inhibitor arcs. As usual, we assume that the sets of places and transitions of a place/transition net are finite.

Terminating Petri nets

To warm up, we first consider the question whether a place/transition Petri net terminates eventually, i.e., whether all its occurrence sequences are finite.

Obviously, a bounded place/transition net terminates if and only if its reachability graph has no cycles. In fact, if the reachability graph has a cycle, then each occurrence sequence from the initial marking to any marking represented by a vertex of the cycle can be extended infinitely, following the arcs of the cycle (remember that each vertex of the reachability graph represents a reachable marking). Conversely, a bounded place/transition net has only finitely many reachable markings, because the set of places of the net is assumed to be finite. Since each occurrence sequence corresponds to a directed path of the reachability graph, each infinite occurrence sequence corresponds to a directed path that passes through at least one vertex more than once; thus the reachability graph has a cycle.

Unbounded Petri nets do not terminate anyway. To see this, consider the construction of the reachability tree. Since the set of transitions is finite, each vertex of this tree has finitely many immediate successors. By König's lemma, the tree has an infinite path, corresponding to an infinite occurrence sequence.

Hence, an obvious algorithm to check termination of a place/transition net first checks boundedness, for example by the coverability graph construction. In case the considered net is bounded, the algorithm constructs the reachabilty graph and checks whether this graph has a cycle. Actually, this two-step approach is not necessary, because the coverability graph of a bounded place/transition net equals its reachability graph and cyclicity of this graph is implicitly checked during the coverability graph construction. A perhaps more elegant algorithm[1] first adds a place to the net which has all transitions of the net in its pre-set and no transition in its post-set, and then checks boundedness of this place, for example by construction of the coverability graph. Obviously, this additional place is bounded if and only if the length of all occurrence sequences is bounded. Since the set of transitions is finite, this is the case if and only if there is no infinite occurrence sequence.

Termination after stopping a transition – the bounded case

We now come back to the question asked initially: Does a place/transition Petri net terminate if a given transition t of the net is stopped eventually? In other words: Is there an infinite occurrence sequence with only finitely many occurrences of t?

For bounded place/transition nets, there is again a very simple algorithmic solution: Construct the reachability graph and check whether every cycle of this graph contains at least one arc labeled by t. In fact, if there is a cycle without t-labeled arc, then – as above – some infinite occurrence sequence starts with a finite sequence to some vertex of this cycle (which might include occurrences of t) and then runs along the cycle infinitely. Conversely, assume that each cycle has at least one t-labeled arc. Each infinite occurrence sequence passes through some vertex of the reachability graph infinitely often. All (infinitely many) subsequences between two subsequent passes through that vertex correspond to a cycle. By assumption all these subsequences contain an occurrence of t, whence t occurs infinitely often in the sequence.

Algorithmically, we can delete all t-labeled arcs in the reachability graph (which does *not* necessarily lead to a connected graph) and check for cycles.

[1] communicated by Karsten Wolf

Dito – the Unbounded Case

Finally, we consider the case that the considered place/transition net is unbounded. Does it eventually terminate provided a given transition t occurs only finitely often? Unfortunately, the coverability graph does not bring immediate help. Consider the simple example of a place/transition net with only one place, which is initially unmarked, a single input transition i, and a single output transition o.

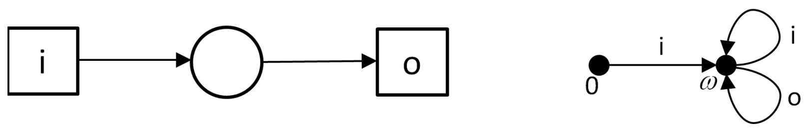

Fig. 1. A simple net and its coverability grap

We say that a transition t eventually stops a net if (and only if) every infinite occurrence sequence contains infinitely many occurrences of t. In the above example, transition i eventually stops the net, whereas transition o does not. However, both transitions occur in the coverability graph in quite the same way, namely as labels of arcs leading from the ω-marking labeled by ω to itself. These are the only cycles of this coverability graph. While the ready coverability graph does thus not lead to an algorithmic solution, we can solve the problem during its construction, as shown below.

Remember that, during the (nondeterministic) construction of the coverability graph, we compare new ω-markings with already constructed ω-markings. An ω-marking is a marking of the places of a net where some places can have the entry ω, meaning that these places can carry arbitrarily many tokens. More precisely, when a new vertex of the coverability graph is constructed, the algorithm compares the ω-marking m corresponding to this new vertex with the ω-markings m' corresponding to vertices which are on paths from the initial vertex to the new one. If, for all places, the new marking m is identical to m', then the new vertex is identified with the vertex corresponding to m'. Otherwise, if $m(s) \geq m'(s)$ for each place s (where $\omega > n$ for every integer n), then m is modified as follows: For each place s with $m(s) > m'(s)$ we set $m(s) := \omega$, because the sequence from the vertex corresponding to m' to the newly constructed vertex can be repeated arbitrarily often, leading to an unbounded token growth on the place s. In the above example, the marking reached by the occurrence of transition i is greater than the initial marking for the only place; hence in the coverability graph this place gets an ω-entry. Further occurrences of transition i are possible, leading to the same ω-marking, because ω already means "arbitrarily many". Notice, however, that transition i can occur infinitely often, no matter if transition o occurs, whereas o cannot occur arbitrarily often without i, and in particular there is no infinite occurrence sequence $o\,o\,o\ldots$, a fact which is not reflected by the coverability graph.

Now we come back to the problem whether some transition t eventually stops its net. To this end, we modify the coverability graph construction as follows. When adding a new vertex and comparing ω-markings with previously reached ω-markings, we also look at the occurrence sequences leading from the previously reached ω-marking to the current one. If all such sequences contain at least one t-transition, we proceed as in the original algorithm. Otherwise we consider the occurrence sequences without t leading from a previously reached ω-marking m' to the actual ω-marking m which satisfy $m'(s) \leq m(s)$ for each place. We define the effect of an occurrence sequence to a place s as the difference between the number of occurrences of output transitions in the sequence and the number of occurrences of input transitions of the sequence. I.e., by the occurrence of the sequence, the token count on s is decreased or increased by the effect of the sequence to s. If $m(s) \neq \omega$ then the effect of the occurrence sequence to s must not be negative by construction. However, if $m(s) = \omega$, then the occurrence sequence might actually decrease the number of tokens on s, as it happens in our example by the short occurrence sequence o.

If we find a sequence (without t) from some suitable previously reached marking m' to m with non-negative effect to all places s, then we stop the algorithm with output *no*, i.e., the algorithm found out that transition t does not stop the net. Otherwise we proceed as in the usual construction of the coverability graph. If the construction algorithm reaches its regular end, i.e., if it never answered *no*, then it delivers the output *yes*, thus identifying that t actually stops the net.

If we apply our modified algorithm to the above trivial example and ask whether o stops the net, then we immediately identify the occurrence sequence $m_0 \xrightarrow{i} m$ which neither contains o nor has a negative effect on any place (but a positive effect on the only existing place). So the algorithm terminates with output *no*. If we apply it with respect to transition i, then the only relevant cycle is given by the arc labeled o, which is actually a loop. The short occurrence sequence o decreases the token count of the only existing place. So it has a negative effect to this place. Therefore, the algorithm finally constructs the complete coverability tree and ends with the output *yes*.

To prove the algorithm correct, we first observe that it proceeds like the usual coverability graph construction algorithm, except that it might terminate earlier. So it terminates eventually, as the unmodified coverability graph construction algorithm terminates eventually.

If the algorithm terminates with ouput *no*, then there is an ω-marking in the coverability graph constructed so far which enables an occurrence sequence without occurrences of t and with non-negative effect to all places. Remember that an ω-marking enables a finite occurrence sequence if the regular marking constructed by replacing all ω-entries by the length of the sequence enables the occurrence sequence (this replacement ensures that none of the transitions of the sequence lacks tokens on places marked by ω). By construction of the coverability graph, we can actually reach such a regular marking by pumping up the tokens on all ω-marked places. Since the occurrence sequence has no negative effect to any place, the marking reached by the sequence assigns at least as many tokens to each place as the marking enabling the sequence. Therefore, the occurrence sequence can be repeated infinitely often. Thus the transition t does not stop the net eventually.

Conversely, assume that a transition t does not stop the net eventually. We proceed indirectly and assume that the algorithm stops with output *yes*, thus constructing the full coverability graph. Since t does not stop the net eventually, there exists a reachable marking m that enables an infinite occurrence sequence without t. In this occurrence sequence, we reach markings m' and m'' (reached after m') such that $m''(s) \geq m'(s)$ for each place s (this is the core of the proof of finiteness of the coverability graph, based on Dickson's Lemma). Let σ be the occurrence sequence leading from m' to m''. Clearly, σ also does not contain t, and it has a non-negative effect to all places. It is known that the ω-markings of the coverability graph cover all reachable markings. Hence some ω-marking m'_ω covers m', i.e., $m'_\omega(s) \geq m'(s)$ for each place s. During the construction of the coverability graph the algorithm will find out that m'_ω enables σ, which leads to another ω-marking m''_ω covering m''. However, comparing m''_ω with m'_ω and considering the occurrence sequence σ would lead to an earlier termination of the algorithm with output *no* – a contradiction.

Conclusion

We have shown how to decide whether a single transition is able to stop an entire net evetually. The proposed algorithm can easily be generalized to sets of transitions (if we forbid all transitions of this set at some time, will the net eventually stop?). Another obvious generalization refers to arc weights; the procedure works for nets with arc weights with only very little changes.

Another tool for identifying transitions that stop a net is given by transition invariants, which are closely related to cyclic occurrence sequences, or by transition sur-invariants, which are related to occurrence sequence with non-negative effect to all places. Both types of invariants can be derived by linear algebraic means. These techniques lead to much more efficient algorithms, but unfortunately provide only sufficient criteria for termination problems.

Please notice that we want to stop nets *eventually*. Applied to human researchers, this means that they should find an end in some years, not immediately. In any case, it is important to find the right transition, i.e., the right way to allow the researchers to concentrate on other beautiful things in live. In the case of Maciej, it might be Martha's job to find the right switch of Maciej, corresponding to such a transition early enough – and vice versa.