Analysis of Recursive Markov Chains, Recursive Markov Decision Processes, and Recursive Stochastic Games

Kousha Etessami
U. of Edinburgh

Joint work with Mihalis Yannakakis, Columbia U.

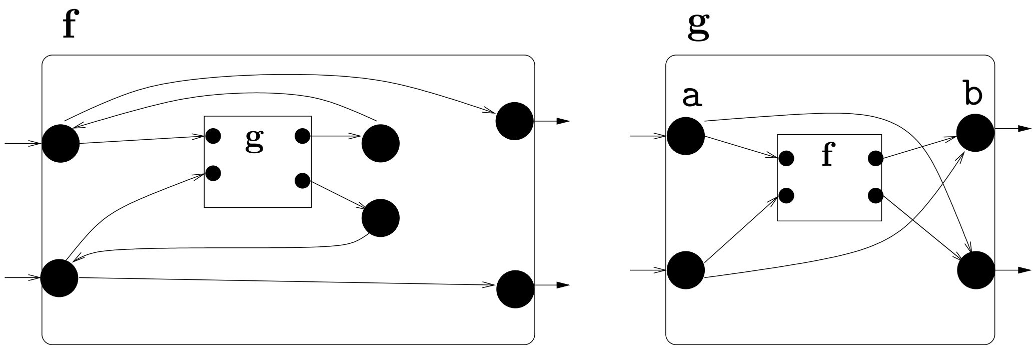
(based on papers at: STACS'05,TACAS'05,ICALP'05,QEST'05, & STACS'06)

Newton Inst. Workshop on Finite and Algorithmic Model Theory, Jan. 9-13, '06

Introduction

- I will be speaking about our work in a recent series of papers with Mihalis Yannakakis on *Recursive Markov Chains (RMCs)* and their extensions.
- There has also been recent research on *probabilistic Pushdown Systems (pPDSs)*, which are closely related to RMCs.
- I will focus on our work on RMCs.
  At the end of my tutorials I will mention the related work and its relation to ours.

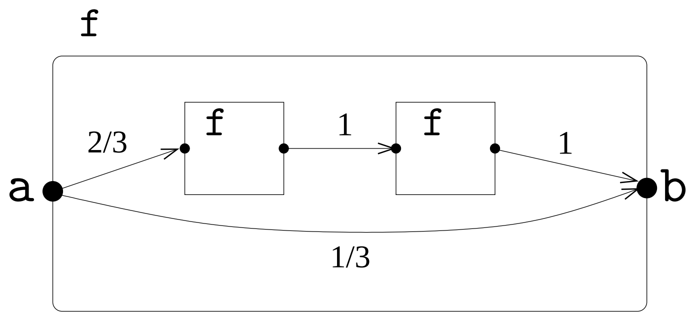
What is a Recursive Graph?



Question: Is it possible to reach **b** from **a**?

This can be computed efficiently: in worst-case cubic time, and in linear time if either the number of entries or exits of each "component" is bounded by a constant.
(More generally, we can check ω-regular properties of recursive graphs with the same model complexity.)
[Alur-E.-Yannakakis'01],[Benedikt-Godefroid-Reps'01], ...

What is a Recursive Markov Chain?

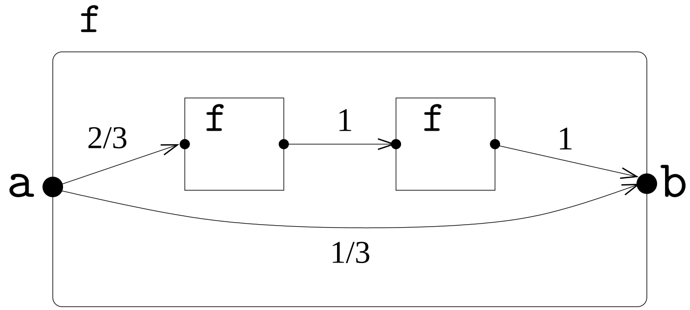


Question: What is the probability of eventually reaching **b** from **a**?
Is there an efficient algorithm for computing such probabilities?

For finite MCs, there's a standard algorithm: solve a linear system of equations associated with the finite MC.

More general model checking question: what is the probability that a run of the RMC satisfies a given ω-regular property (given by a Büchi automaton)?

Let's calculate this termination probability



Let x be the (unknown) probability that starting at $\mathbf{a}$ (in the empty calling context) we will eventually reach $\mathbf{b}$ (in the empty calling context) and terminate.

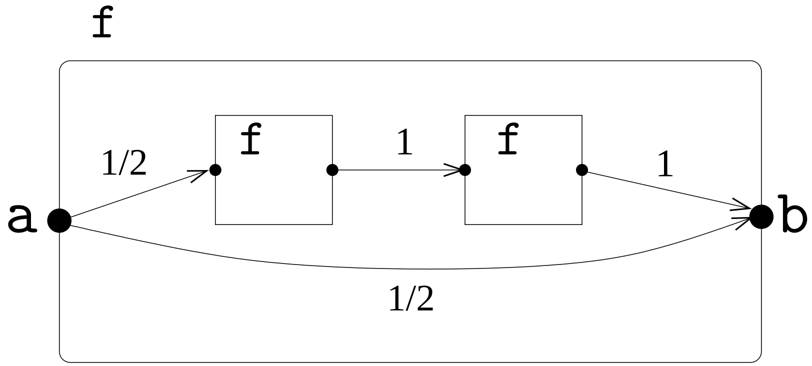
An equation for x: $x = (2/3)x^2 + 1/3$

Note: this is a nonlinear equation. It has two solutions: $x = 1/2$ and $x = 1$.

The *least* solution, let's call it the *Least Fixed Point* (LFP), is: $x^* = 1/2$.

Fact: This is the probability we are after. (For now, trust me!)

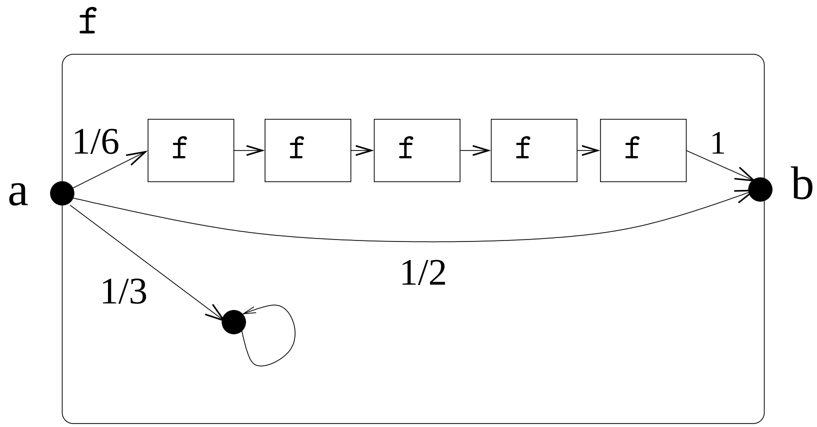
some illustrative examples, part 2: value dependence



Equation: $x = (1/2)x^2 + 1/2$.
Two (degenrate) solutions: $x = 1$ and $x = 1$. LFP: $x^* = 1$.

So, we can have *structurally identical* RMCs, where the termination probability is 1 in one of them (i.e., almost sure termination) but not in the other.
This can't happen with finite Markov Chains.

some illustrative examples, part 3: irrational probabilities

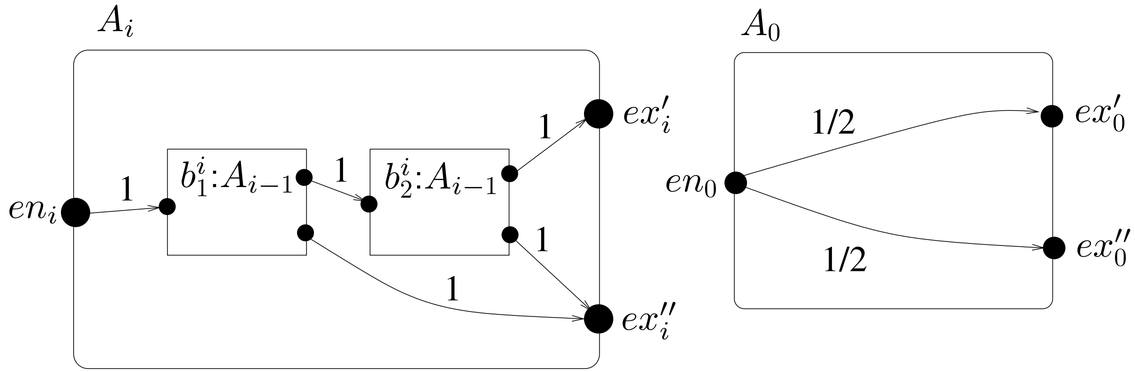


Equation $x = (1/6)x^5 + 1/2$. Thus, $(1/6)x^5 - x + 1/2 = 0$.

This is an irreducible univariate polynomial with "Galois group" S_5. Thus the probability is irrational, and not "solvable by radicals". (approx.: $0.50550123\ldots$)

For finite Markov chains, such probabilities are "concise" rationals.

some illustrative examples, part 4: very small, and very large, probabilities



Fact: $x^*_{(A_n,en,ex'_n)} = \frac{1}{2^{2^n}}$ and $x^*_{(A_n,en,ex''_n)} = 1 - \frac{1}{2^{2^n}}$.

Motivation for studying RMCs

- Recursive Graphs and *Recursive State Machines* (RSMs) ([Alur-E.-Yannakakis'01],[BenediktGodefroidReps'01]) are a natrual abstract model of procedural programs with potential recursion. They are expressively equivalent to Pushdown Systems (PDSs). Lots of theoretical and practical work on such models in verification and program analysis. (Too many references to mention.)

- Recursive Markov Chains (RMCs) arise naturally when we introduce either "intrinsic" or "extrinsic" randomness into such models. Lots of theoretical and practical work on verification and model checking of finite Markov Chains (in both discrete and continuous time, etc.). (See, e.g., [Kwiatkowska,LICS'03] for a recent survey.)

  But RMCs define infinite state Markov chains . . . . . .

analysis of infinite-state probabilistic systems

RMCs define a natural class of denumerable Markov Chains that generalize several important classes of stochastic processes.

RMCs generalize *Stochastic Context-Free Grammars*(SCFGs), studied since the 1970's in the Natural Language Processing community, and more recently in biological sequence analysis.

RMCs also generalize *Multi-Type Branching Processes* (MT-BPs).
([Galton-Watson,1874],[Kolmogorov-Sevastyanov'48], [Everett-Ulam'48],[Harris'63],. . .).
Branching processes are important stochastic processes, with many applications.

MT-BPs and SCFGs are closely related, and both correspond to the restricted class of "single-exit"-RMCs.

Despite the extensive research on MT-BPs and SCFGs, basic algorithmic question about them remained unanswered, not to mention the more general RMCs.

RMCs, more formally

An RMC, $A = \langle A_1, \ldots, A_k \rangle$ consists of **components** $A_1, \ldots, A_k$, with each A_i given by:

- A set N_i of **nodes**, and a set B_i of **boxes**.
  A mapping $Y_i : B_i \mapsto \{1, \ldots, k\}$ of each box to a component.
- A set $En_i \subseteq N_i$ of **entry nodes**, and a set $Ex_i \subseteq N_i$ of **exit nodes**.
- A **transition relation** δ_i, where each $(u, p_{u,v}, v) \in \delta_i$ has the form:
  - $u \in N_i$, or $u = (b, ex)$ where $b \in B_i$ and $ex \in Ex_{i_b}$.
  - $v \in N_i$, or $v = (b, en)$ where $b \in B_i$ and $en \in En_{i_b}$.
  - $p_{u,v} \in \mathbb{R}_{\geq 0}$,
    and $\sum_v p_{u,v} = 1$ or $= 0$.
    (The sum ranges over all "vertices" v to which u has a transition.)

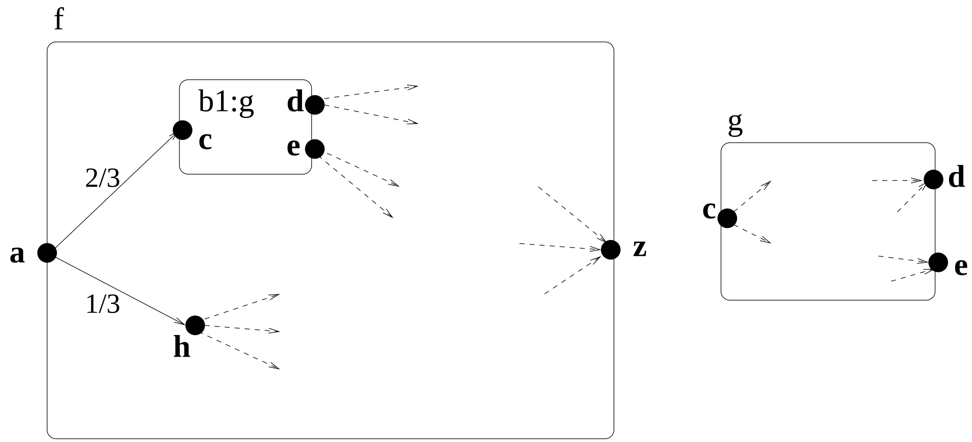
The underlying global markov chain of an RMC

- *Expanding* an RMC, A, defines a "global" countable markov chain, $\mathcal{M}_A$.
- **States** of $\mathcal{M}_A$ have the form $s = \langle b_1 b_2 \ldots b_r, x\rangle$, where b_i's are boxes (the "context" or "call stack"), and x is a node.
- Transitions $(s, p_{s,s'}, s')$ of $\mathcal{M}_A$ are dictated in the obvious way by transitions of A.

Key Question: For a vertex u and an exit ex, both of some component A_i in a RMC, what is the probability of eventually reaching (and terminating at) the global exit state $\langle \epsilon, ex\rangle$ of $\mathcal{M}_A$ starting at the global state $\langle \epsilon, u\rangle$?

Let us denote this (unknown) probability by $x_{(i,u,ex)}$.
(Using these probabilities, we can also calculate other reachability probabilities.)

The non-linear system associated with an RMC



What is $x_{(f,\mathbf{z},\mathbf{z})}$? $\quad x_{(f,\mathbf{z},\mathbf{z})} = 1$

What is $x_{(f,\mathbf{a},\mathbf{z})}$? $\quad x_{(f,\mathbf{a},\mathbf{z})} = \frac{1}{3} x_{(f,\mathbf{h},\mathbf{z})} + \frac{2}{3} x_{(f,(b1,\mathbf{c}),\mathbf{z})}$

What is $x_{(f,(b1,\mathbf{c}),\mathbf{z})}$? $\quad x_{(f,(b1,\mathbf{c}),\mathbf{z})} = x_{(g,\mathbf{c},\mathbf{d})} x_{(f,(b1,\mathbf{d}),\mathbf{z})} + x_{(g,\mathbf{c},\mathbf{e})} x_{(f,(b1,\mathbf{e}),\mathbf{z})}$

These "patterns" cover all cases, in general yielding a system of polynomial equations of the form:

$$\bar{x} = P(\bar{x})$$

Basic facts about the system $\mathbf{x} = P(\mathbf{x})$

- The coefficients in $P()$ are non-negative, and in fact $P : \mathbb{R}^n \mapsto \mathbb{R}^n$ defines a monotone operator on $D \subseteq [0,1]^n$.

  By a Tarski-Knaster argument, $P()$ has a Least Fixed Point $\mathbf{x}^*$ in $[0,1]^n$.

  For a vector x, let $P^m(x) = P(P(....(P(x))))))$, i.e., m iterations of the operator $P()$.

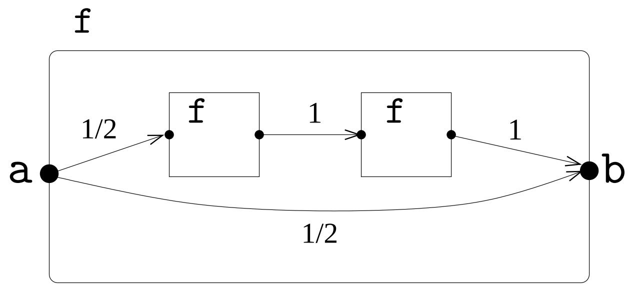
- **Theorem 0:** The LFP, $x^* = \lim_{m\to\infty} P^m(\mathbf{0})$, gives precisely the termination probabilities we are after. ($x^* = P(x^*)$ and x^* is the least nonnegative solution of $x = P(x)$.)

- Can we compute these probabilities efficiently?

some illustrative examples, part 5: exponentially many standard iterations required

Question: How many iterations m of $P^m(0)$ are required to obtain the probabilities to within i bits of precision?

Answer: In the worst case, at least exponentially many iterations (in i), even for a fixed RMC:



Equation: $x = (1/2)x^2 + 1/2$.

Fact: LFP $x^* = 1$, but for $m \leq 2^i$, $|1 - P^m(0)| \geq 1/2^i$.

RMCs and the Existential Theory of the Reals

A sentence in the first-order theory of reals looks something like this:

$$\exists x_1, x_2 \forall x_3 (f_1(\bar{x}) \geq 0 \wedge f_2(\bar{x}) < 0) \vee f_3(\bar{x}) = 5$$

where $f_i(\bar{x})$'s are multi-variate polynomials. Existential (prenex) sentences look like:

$$\exists x_1, \ldots, x_k B(\bar{x})$$

where $B(\bar{x})$ is a boolean combination of "polynomial predicates".

Building on 60 years of work since Tarski's quantifier elimination procedure ([Tarski'51,Collins'70's,Canny'89,Renegar'92,Basu-et.al.'96,...]), we know we can decide the truth of an $\exists$-theory formula in PSPACE, and in time that is exponential only in the number of variables.

Using the ∃-theory

For our system $\bar{x} = P(\bar{x})$, let $f_i(\bar{x}) \equiv P_i(\bar{x}) - x_i$.

Thus, if we want to check whether the LFP vector is "below" another vector $\bar{c}$, we just have to check:

$$\varphi \equiv \exists x_1, \ldots, x_n \bigwedge_{i=1,\ldots,n} f_i(x_1, \ldots, x_n) = 0 \wedge \bigwedge_{i=1,\ldots,n} 0 \leq x_i \leq c_i$$

Hence, by a "binary search" on each coordinate, we can obtain the LFP value in that coordinate to within i bits of precision with $O(i)$ queries to the ∃-theory.

Theorem 1. *In PSPACE we can decide whether each probability x_i^* is above/equal/below a given rational value $p \in [0, 1]$, and we can also "approximate" the LFP, x^*, to any given number of bits, j (in unary), of precision.*

If we restrict the class of RMCs, we can do better.....

Special classed of RMCs: bounded RMCs

We call an RMC *bounded* if the total number of entries and exits of all components is bounded by a constant. But the number of nodes is not restricted.

Theorem 2. *For a bounded RMC, we can decide in polynomial time whether each* x_i^* *is above/equal/below a given rational value* $p \in [0, 1]$*, as well as approximate the probability to* j *bits of precision.*

The proof is non-trivial. We use the ∃-theory results, but we also need other techniques, in particular, a rational function representation on a bounded number of variables for the probabilities in question.

Single-exit RMCs and Branching Processes

Single-exit RMCs are RMCs where each component has exactly one exit. (But there may be many components, each with many entries.)

1-exit RMCs are "equivalent to" Stochastic Context-Free Grammars. As mentioned, SCFGs are intimately related to MT-BPs [Kol-Sev'48,Ev-Ul'48].

Results about Multi-Type BPs [Sevastyanov'48-51,Harris'63] allow us to characterize whether the "probability of extinction" is exactly 1 based on the eigenvalues and spectral radius of certain matrices, namely the Jacobian of a decomposition of $x = P(x)$, evaluated at the all 1 vector.
We give a "modern" proof of these results, and we use it to show:

Theorem 3. *There is a P-time algorithm to determine, for each reachability probability, x_i^*, of a single-exit RMC, whether:*
(a) $x_i^* = 0$, *(b)* $x_i^* = 1$, *or (c)* $0 < x_i^* < 1$.

Linearly-recursive RMCs

Definition We call an RMC *linearly recursive* (`lr`-RMC) if there is no path of transitions, inside any component, from some box-exit to some box-entry.

Theorem *For* `lr`*-RMCs, there is a P-time algorithm for calculating the* exact *(rational) termination probabilities,* x_i^**.*

We give a much more general algorithm and result, from which the above theorem follows as a very special case.

Namely, we show that a "decomposed Newton's method" converges monotonically to the LFP of the systems x=P(x) for RMCs.

RMCs and the Square-Root Sum problem

The square root sum problem is the following decision problem: given $(d_1, \ldots, d_n) \in \mathbb{N}^n$ and $k \in \mathbb{N}$, decide whether $\sum_{i=1}^{n} \sqrt{d_i} \leq k$.

It is known to be solvable in PSPACE, but it has been a major open problem since the 1970's ([GareyGrahamJohnson'76]) whether it is solvable even in NP. It has important consequences in subjects such as computational geometry.

Theorem 4 *The square-root sum problem is polynomial-time reducible to the problem of, given a single-exit RMC A, and given a rational p, determining whether $x^*_{(1,en,ex)} \geq p$.*

Moreover it is also polynomial time reducible to the problem of determining whether $x^*_{(1,en,ex_1)} = 1$ in a 2-exit RMC.[1]

[1]This latter fact was also observed by Esparza and Kucera (2004), based on a preliminary draft of our tech. report for the STACS'05 paper, which stated and proved the first part of the theorem.

Newton's method

(Multi-variate) Newton's method aims to find a solution of $F(\mathbf{x}) = 0$, by starting at some initial guess vector $\mathbf{x}_0$, and computing the sequence $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \ldots$, where:

$$\mathbf{x}_{k+1} := \mathbf{x}_k - (F'(\mathbf{x}_k))^{-1} F(\mathbf{x}_k)$$

Here $F'(\mathbf{x})$, is the *Jacobian matrix*, of partial derivatives, given by

$$F'(\mathbf{x}) = \begin{bmatrix} \frac{\partial F_1}{\partial x_1} \cdots \frac{\partial F_1}{\partial x_n} \\ \vdots \vdots \vdots \\ \frac{\partial F_n}{\partial x_1} \cdots \frac{\partial F_n}{\partial x_n} \end{bmatrix}$$

general facts about Newton's method

- The method won't even be defined unless the matrix $F'(\mathbf{x}_k)$ is non-singular for all k.
- Even if it is defined, it can diverge, and in fact diverges even for some degree 3 univariate polynomials.
- But when it does converge, it is typically very fast....
- Remarkably, we show that for RMCs, Newton's method, when suitably decomposed, converges monotonically to the LFP solution.

RMCs and Newton's method

Let $F(\mathbf{x}) = P(\mathbf{x}) - \mathbf{x}$.

Once we decompose $x = P(x)$ in a natural way into its *Strongly Connected Components* (SCCs), then

Theorem 6 *The Decomposed Newton's Method, started at* $\bar{0}$ *on the decomposition of* $x = P(x)$*, "monotonically converges" to the LFP. meaning that* $(F'(x_k))^{-1}$ *exists for all* $k \geq 0$*, and* $0 = \mathbf{x}_0 \leq \mathbf{x}_1 \leq \mathbf{x}_2 \leq \ldots \leq \mathbf{x}^*$*, and* $\lim_{k\to\infty} \mathbf{x}_k = \mathbf{x}^*$*.*
Moreover, for all $k \geq 0$*,* $\mathbf{x}_k \geq P^k(0)$*.*

This is a very useful result for practical numerical computation of these probabilities.

Decomposed Newton's method and linearly-recursive RMCs

Proposition *Our decomposed Newton's method, applied directly to the system* $\mathbf{x} = P(\mathbf{x})$ *for* `lr`*-RMCs computes the* exact *rational termination probabilities* $\mathbf{x}^*$ *in P-time.*

More generally, it computes such exact rational values for all *piecewise linearly-recursive* RMCs, where the strongly-connected components of the (non-linear) system $\mathbf{x} = P(\mathbf{x})$ encountered bottom-up (in the DAG of SCCs) are all linear.

Ok, now on to model checking

Given a labeled RMC, A, and a Büchi automaton B, let $P_A(L(B))$ denote the probability that an exectution of A is in the ω-language $L(B)$.

We are interested in the following two kinds of problems:

(1) The *qualitative* model checking problems:
Is $P_A(L(B)) = 1$? Is $P_A(L(B)) = 0$?

(2) The *quantitative* model checking problems: given $p \in [0, 1]$, is $P_A(L(B)) \geq p$? Also, we may wish to approximate $P_A(L(B))$ to within a given number of bits of precision.

Our results on ω-regular model checking of RMCs

Theorem:

Qualitative:

	reachability	det. Büchi	nondet. Büchi
1-exit	P	P	P in RMC, EXPTIME in Büchi
Bounded	P	P	P in RMC, EXPTIME in Büchi
General	PSPACE	PSPACE	PSPACE in RMC, EXPTIME in Büchi

Quantitative:

	reachability	det. Büchi	nondet. Büchi
1-exit	PSPACE	PSPACE	PSPACE in RMC, EXPSPACE in Büchi
Bounded	P	P in RMC for fixed Büchi	P in RMC, for fixed Büchi
General	PSPACE	PSPACE	PSPACE in RMC, EXPSPACE in Büchi

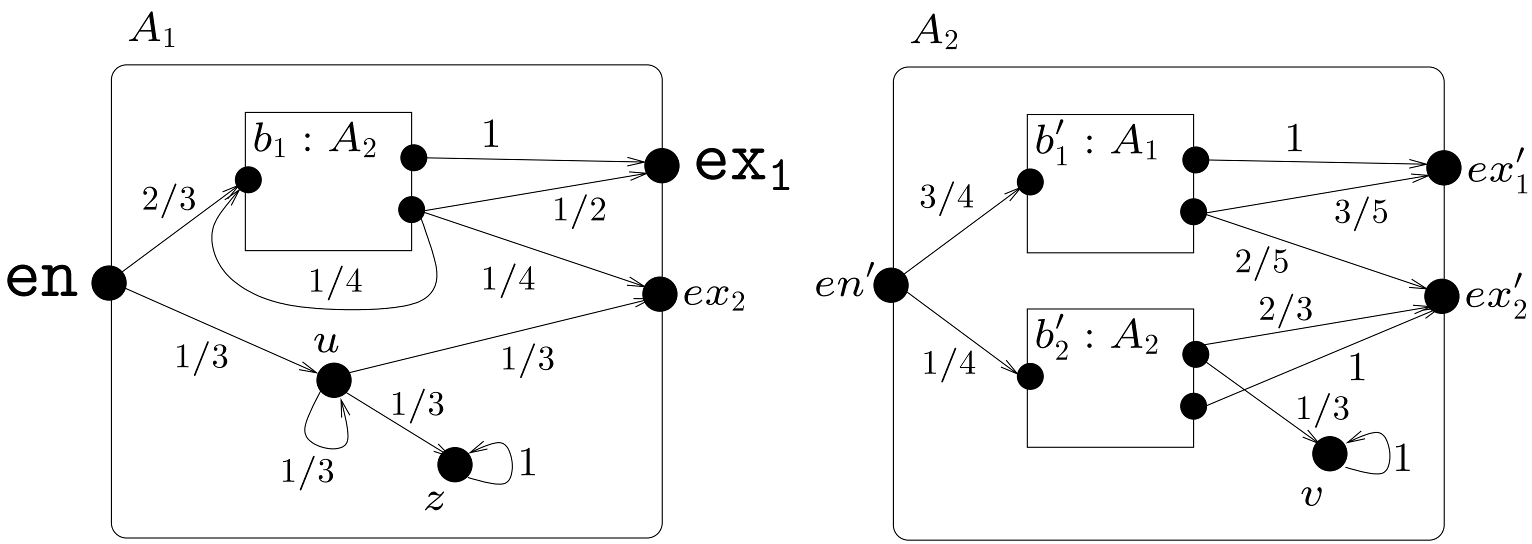
Moreover......

Theorem 7 *Qualitative model checking, against a non-det. Büchi automaton, is EXPTIME-hard (thus EXPTIME-complete), even for a fixed 1-exit 1-entry RMC.*

Brief hints of the many techniques involved

- A finite *conditioned summary chain*, $\mathcal{M}'_A$ can be "built" using the "reachability" solution probabilities x^*.
  This extends our summary graph construction for RSMs from [Alur-E.-Yannakakis'01] (see also [BGR'01]) to the probabilistic setting.
  We show that there is a "*probability preserving transformation*" from the infinite MC, $\mathcal{M}_A$, to the finite conditioned MC, $\mathcal{M}'_A$.
- Many extensions of techniques from [Courcoubetis-Yannakakis'95], in particular allowing us to avoid full-fledged "determinization" of non-det. Büchi automata.
- A crucial *unique fixed point theorem*.
- EXPTIME lower bound: reduction from Alt.-Linear-Space Turing Machines.
- Our upper bounds for Bounded RMCs involve rational function characterizations of probabilities, building on our reachability work.

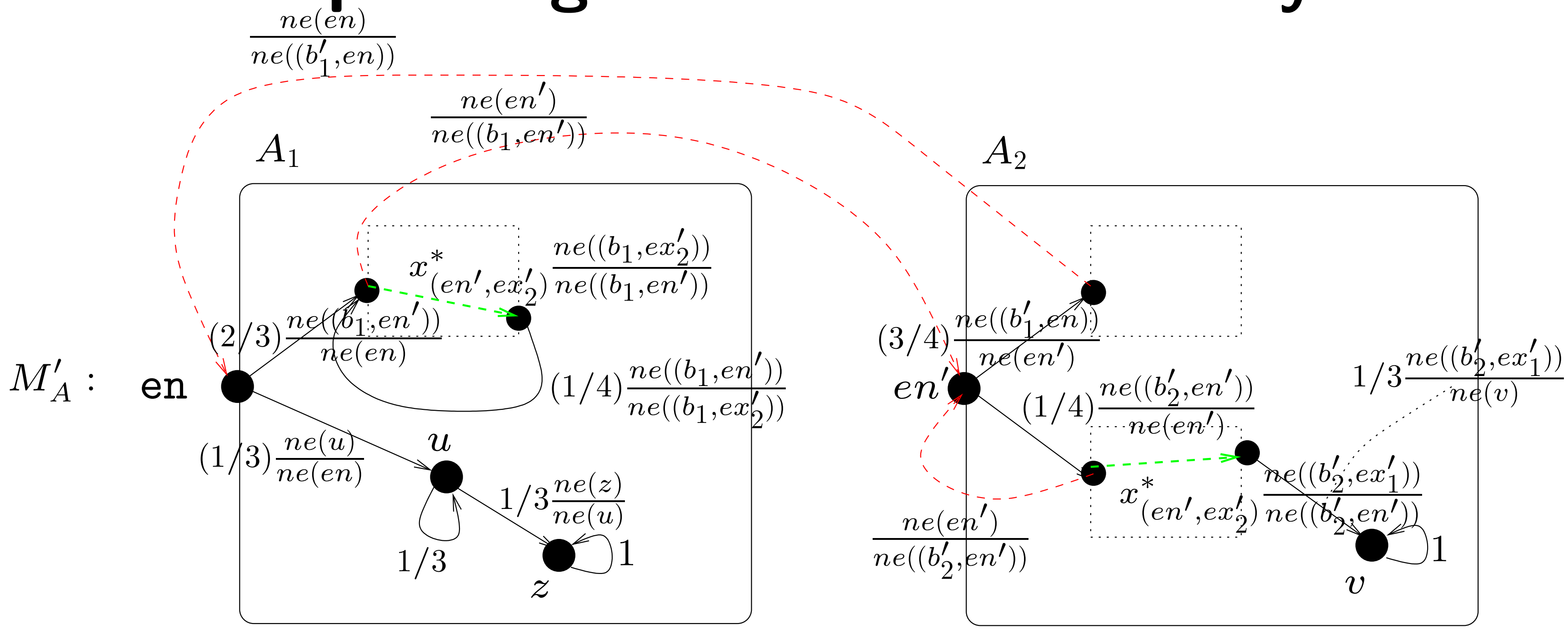
Example: an RMC



Let x^* be the LFP solution to $x = P(x)$ for this RMC.

For a vertex u in A_i, let $ne(u) = 1 - \sum_{ex \in Ex_i} x^*_{(u,ex)}$, be the probability of never exiting the component when starting at u.

The corresponding conditioned summary chain



Each transition probability is now precisely the conditional probability of making that transition, given that you will never exit the component in question. Now, e.g.: $P_A(\Box\Diamond v)$ = probability of reaching bottom SCC containing v in M'_A.

The Unique Fixed Point Theorem

Another key to our model checking results is the following fundamental theorem:

Call a vertex u of an RMC "deficient" if $ne(u) > 0$, i.e., starting at u, there is a chance that we will never exit its component.

Theorem 8 (*The Unique Fixed Point Theorem*) *The system $x = P(x)$ has a unique solution that satisfies $\sum_{ex} x_{(u,ex)} < 1$ for every deficient vertex u, and $\sum_{ex} x_{(u,ex)} = 1$ for every other vertex u.*

This generalizes a 50 year old result on Branching Processes ([Sevastyanov'48-51], [Harris'63]).

It is a highly non-trivial generalization.

The theorem allows us to characterize the LFP of $x = P(x)$ uniquely, using only the *Existential Theory of Reals*.

What about LTL properties instead of Büchi?

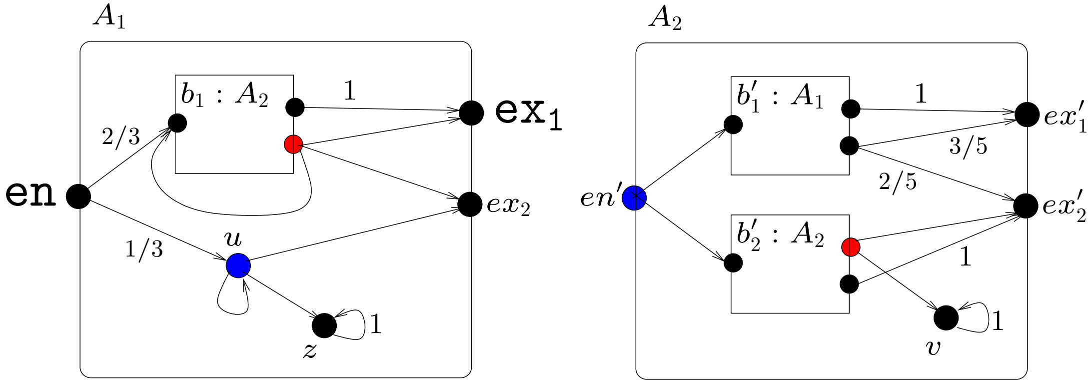
Theorem [Yannakakis-E.,QEST'05] *All of the complexities in the table remain the same, as well as the EXPTIME-hardness, when you replace properties specified by non-det. Büchi automata with properties specified by LTL formulas.*

This might surprise you: LTL formulas can be exponentially more compact than non-det. Büchi automata. But those who know the [Courcoubetis-Yannakakis'89] results for model-checking finite Markov chains won't be so surprised.

Markov chains, and RMCs, are one of the only settings where the [Vardi-Wolper'86] automata-theoretic approach to LTL model checking is suboptimal.
The proof of the above theorem uses the intricate results in [C-Y'89], plus a number of non-trivial extensions to them. One crucial fact is that LTL properties can be expressed by reverse-deterministic Büchi automata.

Ok, now let's extend RMCs to RMDPs and RSSGs

Recursive Markov Decision Processes (RMDPs) are a natural extension of RMCs, where some nodes are *controlled* (by Player 1), while others are probabilistic. *Recursive Simple Stochastic Games* (RSSGs) extend RMDPs: some nodes are controlled by Player 1, others by Player 2, and the rest are probabilistic.



RMDPs (and to a lesser extend, RSSGs) are useful for modeling non-deterministic behavior, as well as modeling a system's interactions with an environment. RSSGs strictly generalize Condon's finite-state Simple Stochastic Games.

some general motivation

- MDPs are a fundamental formalism for control optimization problems in sequential stochastic environments, and have had many applications.
- Both MDPs and Stochastic Games have a vast literature, dating back to Bellman and Shapley in the 1950s.
- MDPs are typically equipped with a reward function and studied under various "reward criteria" (e.g., "discounted reward", etc.).
- We will consider the more basic reachability and termination questions without rewards (and also model checking questions).
- Finite MDPs are already used in model checking tools for probabilistic systems, such as PRISM (Birmingham).

The global infinite SSG

An RSSG A defines a denumerable SSG $M_A = (V, \Delta, \mathrm{p}layer)$. Let Ψ_i denote the set of strategies for player i. A pair of strategies $\sigma \in \Psi_1$ and $\tau \in \Psi_2$ induces a countable Markov chain $M_A^{\sigma,\tau} = (V^*, \Delta')$, whose states are *histories* in M_A.

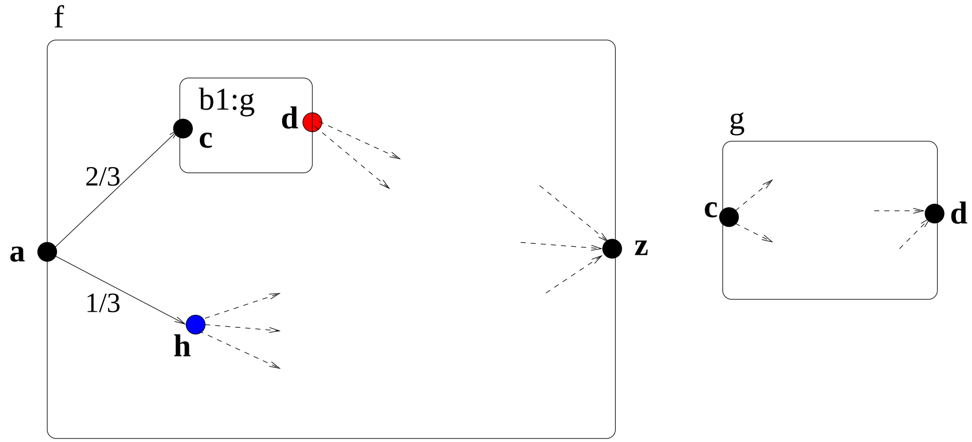
Let $x^{*,\sigma,\tau}_{(u,ex)}$ be the probability of terminating at $w\langle \epsilon, ex\rangle$, for some $w \in V^*$, starting at $\langle \epsilon, u\rangle$, in $M_A^{\sigma,\tau}$. Let $x^*_{(u,ex)} = \sup_{\sigma\in\Psi_1} \inf_{\tau\in\Psi_2} x^{*,\sigma,\tau}_{(u,ex)}$.

It follows from general determinacy results, e.g., Tony Martin's (1998) "Blackwell Determinacy" which applies to all Borel zero-sum stochastic games with a countable state-space, that RSSG termination games are **determined**, i.e.:

$$\sup_{\sigma\in\Psi_1} \inf_{\tau\in\Psi_2} x^{*,\sigma,\tau}_{(u,ex)} = \inf_{\tau\in\Psi_2} \sup_{\sigma\in\Psi_1} x^{*,\sigma,\tau}_{(u,ex)}.$$

Central Algorithmic Problem: Calculate the value $x^*_{(u,ex)}$ of the termination game (starting at u and terminating at ex).

1-exit RSSGs and nonlinear min/max equations



What is $x_{(f,\mathbf{z},\mathbf{z})}$? $\quad x_{(f,\mathbf{z},\mathbf{z})} = 1$

What is $x_{(f,\mathbf{a},\mathbf{z})}$? $\quad x_{(f,\mathbf{a},\mathbf{z})} = \frac{1}{3}x_{(f,\mathbf{h},\mathbf{z})} + \frac{2}{3}x_{(f,(b1,\mathbf{c}),\mathbf{z})}$

What is $x_{(f,(b1,\mathbf{c}),\mathbf{z})}$? $\quad x_{(f,(b1,\mathbf{c}),\mathbf{z})} = x_{(g,\mathbf{c},\mathbf{d})}x_{(f,(b1,\mathbf{d}),\mathbf{z})}$

What is $x_{(f,\mathbf{h},\mathbf{z})}$? $\quad x_{(f,\mathbf{h},\mathbf{z})} = \max_{\{\text{neighbors } \mathbf{v} \text{ of } \mathbf{h}\}} x_{(g,\mathbf{v},\mathbf{z})}$

What is $x_{(f,(b1,\mathbf{d}),\mathbf{z})}$? $\quad x_{(f,(b1,\mathbf{d}),\mathbf{z})} = \min_{\{\text{neighbors } \mathbf{v} \text{ of } (b1,\mathbf{d})\}} x_{(g,\mathbf{v},\mathbf{z})}$

We get a new system of the form $\bar{x} = P(\bar{x})$.

Facts about the 1-exit system $\mathbf{x} = P(\mathbf{x})$ (déjà vu)

- The coefficients in $P()$ are non-negative, and in fact $P : \mathbb{R}^n \mapsto \mathbb{R}^n$ defines a monotone operator on $[0,1]^n$.

  By a Tarski-Knaster argument, $P()$ has a Least Fixed Point $\mathbf{x}^*$ in $[0,1]^n$.

- **Theorem** The LFP of $x = P(x)$, namely the vector $x^* = \lim_{m \to \infty} P^m(0)$, gives precisely the values of the games starting at each vertex.

  (This is actually somewhat more subtle to prove than for RMCs.)

- Can we compute these values effectively? Yes.

1-exit RSSGs and the ∃-theory

Theorem For a 1-exit RSSG, we can decide whether $x^*_{(u,ex)} \geq p$, for a given rational p, in PSPACE, and we can also approximate the probabilities to within i bits of precision in PSPACE.

Proof Again, we use the ∃-theory of Reals. Simply note that:

$$z = \max_i x_i \iff (\bigwedge_i z \geq x_i \wedge \bigvee_i z = x_i)$$

Likewise: $z = \min_i x_i \iff (\bigwedge_i z \leq x_i \wedge \bigvee_i z = x_i)$

So, for $c \in \mathbb{R}^n$, we can again write the following ∃-theory sentence:

$$\varphi \equiv \exists x_1, \ldots, x_n \bigwedge_{i=1,\ldots,n} x_i = P_i(x_1, \ldots, x_n) \wedge \bigwedge_{i=1,\ldots,n} 0 \leq x_i \leq c_i$$

which is true if and only if $\mathbf{x}^* \leq c$. We can likewise do a "binary search" on each coordinate using the ∃-theory. ∎

S&M Determinacy and 1-exit RSSGs

For 1-exit RMCs, we can generalize Condon's (1989) memoryless determinacy result for finite SSGs in a very strong sense.

Definition *A strategy in an RSSG termination game is called* **Stackless and Memoryless** *(***S&M***), if it neither depends on the history nor on the current calling context (or call stack).*

The game is called **S&M-determined** *if one player or the other has an S&M value-achieving strategy (regardless of what the other player does).*

Theorem *1-exit RSSG termination games are S&M-determined.*

Although the statement is intuitive, our proof is *very* delicate, relying on properties of certain power series that arise when studying 1-exit RSSGs.

Some immediate consequences

Corollary

1. *For maximizing 1-exit RMDPs, we can decide qualitative termination, i.e., whether $x_i^* = 1$, in NP.*

2. *For minimizing 1-exit RMDPs, we can decide qualitative termination in coNP.*

3. *For 1-exit RSSGs, we can decide qualitative termination in $\Sigma_P^2 \cap \Pi_P^2$.*

Proof: e.g., of (1): Just "guess" a S&M strategy for maximizer, and verify that in the resulting 1-exit RMC, $x_i^* = 1$, by using our earlier algorithm for qualitative termination of 1-exit RMCs. ∎

Can we do better? You bet!

P-time qualitative analysis of 1-exit RMDPs

Theorem ([E.-Yannakakis,STACS'06])

1. *For both maximizing and minimizing 1-exit RMDPs the qualitative termination problem can be answered in P-time.*
2. *For 1-exit RSSGs, the qualitative termination problem is in NP$\cap$coNP.*

We give distinct algorithms for max- and min- RMDPs (they are not symmetric). The algorithms and proofs combine Perron-Frobenius theory together with an iterative algorithm and Linear Programming formulation of (joint) spectral radius characterizations of both maximizing and minimizing 1-exit RMDP qualitative termination problems.....
network flows also come into play.....
(Some techniques were inspired by recent work of [Denardo-Rothblum'05], on multi-matrix multiplicative systems.) ∎

Can we do better than NP∩coNP for 1-exit RSSGs?
Not without a major breakthrough.

The qualitative termination problem for 1-exit RSSGs is at least as hard as Condon's quantitative termination problem for finite SSGs:

Theorem([E.-Yannakakis'06]) *There is a P-time reduction from Condon's* quantitative *termination problem for finite SSGs (namely is* $x_i^* \geq 1/2$*?) to the qualitative termination problem for 1-exit RSSGs.*

The proof is not difficult. It exploits RMCs that are "structurally identical", but disagree on termination with probability 1. ■

It is not at all clear that there is a reduction in the other direction.

Note that the harder quantative 1-exit RSSG termination problem, is also at least as hard as the Square-Root Sum problem.

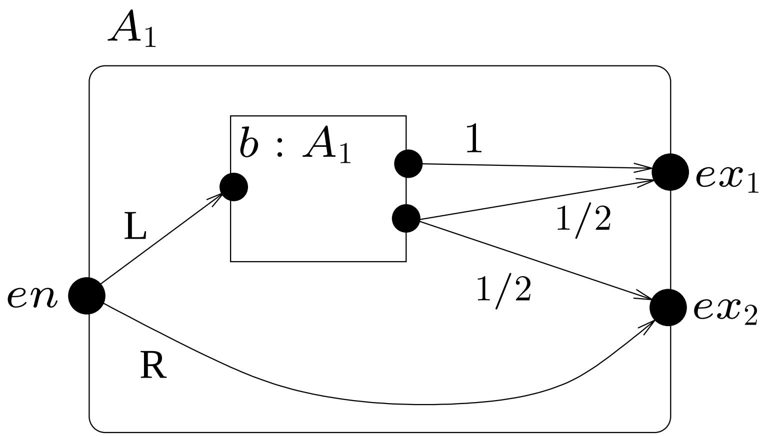
As is known, for finite SSGs the qualitative problem is easy to compute (in P-time). In fact, more generally, we show:

Theorem

1. *For "linearly-recursive" 1-exit RMDPs the exact, rational, value of the game can be computed in P-time.*

2. *For linearly-recursive 1-exit RSSGs, the qualitative termination problem is decidable in P-time.*
   (Note, this generalizes the same fact for finite SSGs.)

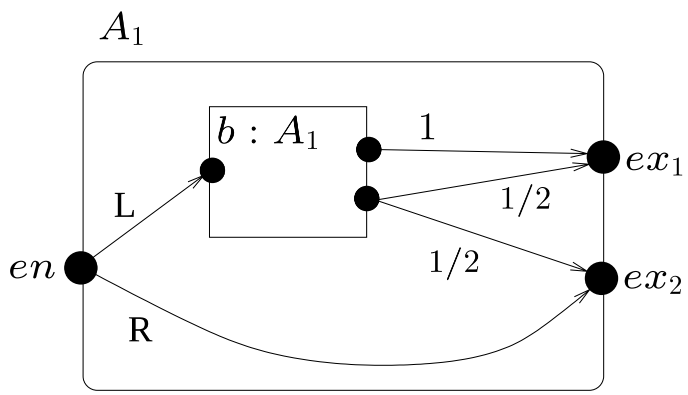
Ok, what about RMDPs with 2 exits?



Question: Consider this 2-exit RMDP.

Starting at en, what would your strategy be to maximize the probability of terminating at ex_1?

There is no optimal strategy.



The strategy $L^n R$ has payoff $(1 - \frac{1}{2^n})$, but no strategy achieves the value 1 of the game. (Note that the S&M strategies both have payoff 0, so they are terrible). The best we can do are ϵ-optimal strategies.

But we can still ask, and hope to answer, whether the value of the game is above/below a given threshold.

Multi-exit RMDPs and RSSGs: Undecidability

Theorem *Given a multi-exit (linearly-recursive) RMDP, it is undecidable whether* $x^*_{(u,ex)} = 1$*, even when the number of exits is bounded by a fixed constant.*

Moreover, for each constant $\epsilon > 0$*, it is not even decidable to distinguish whether* $x^*_{(u,ex)}$ *is 1 or is* $< \epsilon$*.*

We prove this via a reduction from **Probabilistic Finite Automata** (**PFA**), [Rabin'63], and [Paz'71, Condon-Lipton'89,.....].
The language emptiness problem for PFAs is to decide, given p, $0 < p < 1$, whether there exists a string accepted by the PFA with probability $\geq p$.
PFA emptiness is undecidable in strong ways.
We show that PFAs are a special case of linearly-recursive RMDPs.

We employ a recent result on PFA emptiness: [Blondel-Canterini'2003]: it is undecidable even when the number of states is constant (46 is enough).

Consequences for model checking RMDPs

Unfortunately, as one consequence we can show:

Theorem *The qualitative model checking problem, for properties specified by LTL or by Büchi automata, even for 1-exit RMDPs, is undecidable. This is so even for a fixed property.*

The reason is because we can essentially encode the PFA as a "product of" a Büchi automaton or LTL formula, and a 1-exit RMDP.

Related work

- [Esparza-Kucera-Mayr'04] studied decidability of model checking for probabilistic Pushdown Systems (pPDSs) against both linear and branching time properties.
  They showed, e.g., decidability of model checking pPDSs against a deterministic Büchi automaton specification (with very high complexity).
- Independently, in [Etessami-Yannakakis,STACS'05] (Tech. Report June'04) we studied RMCs, but we considered reachability analysis only.
  There we obtained our strong upper and "lower" bounds for both qualitative and quantitative reachability analysis on RMCs.
- [Brazdil-Kucera-Strazovsky'05] then studied model checking of pPDSs further, in particular, against general nondeterministic Büchi specifications.
  They showed that, e.g., quantitative model checking can be decided in overall 3-EXPTIME, and in EXPTIME in the size of the pPDS model only.

- In [E.-Yannakakis, TACAS'05] we extended our study to ω-regular model checking of RMCs, yielding substantial complexity improvements (exponential or more in several setting) over both [Esparza-Kucera-Mayr'04] and [Brazdil-Kucera-Strazovsky'05], when translated to the setting of pPDS.
- In [E.-Yannakakis,ICALP'05] we extended our study of RMCs to RMDPs and RSSGs. The results for qualitative analysis of 1-RMDPs and 1-RSSGs were strengthened substantially in [STACS'06].
- In [Yannakakis-E., QEST'05], we extended our model checking study to the case of LTL specifications.
- [Brazdil, Esparza, Kucera, et. al.'05] have also done recent work on computing expectation and variance under reward functions for pPDSs.

More related work

- [Fagin,Karlin,Kleinberg,Raghavan,Rajagopalan,Rubinfeld,Sudan,'01] studied what they called "Random Walks with Back-Buttons", as a probabilistic model of surfing and "web crawling" on the WWW. Their model corresponds to a strict subclass of 1-exit RMCs. They showed that for their model they can approximate termination probabilities and approximate *Cesaro limit distributions* on states in P-time, using semi-definite programming.

- [Jha,Reps,'04] [SchwoonJhaRepsStubblebine'03] use *weighted*-Pushdown Systems (with weights over the $\{\max, +\}$ algebra), to analyze the security of certificate chains in the SPKI/SDSI public key infrastructure. (Our techniques may enable some extensions of their work.)

potential applications in computational biology and population biology

As we said, RMCs generalize several important classes of stochastic processes.

- RMCs generalize *Multi-Type Branching Processes* (MT-BPs). ([Kolmogorov-Sevastyanov'48],[Everett-Ulam'48],[Harris'63]).

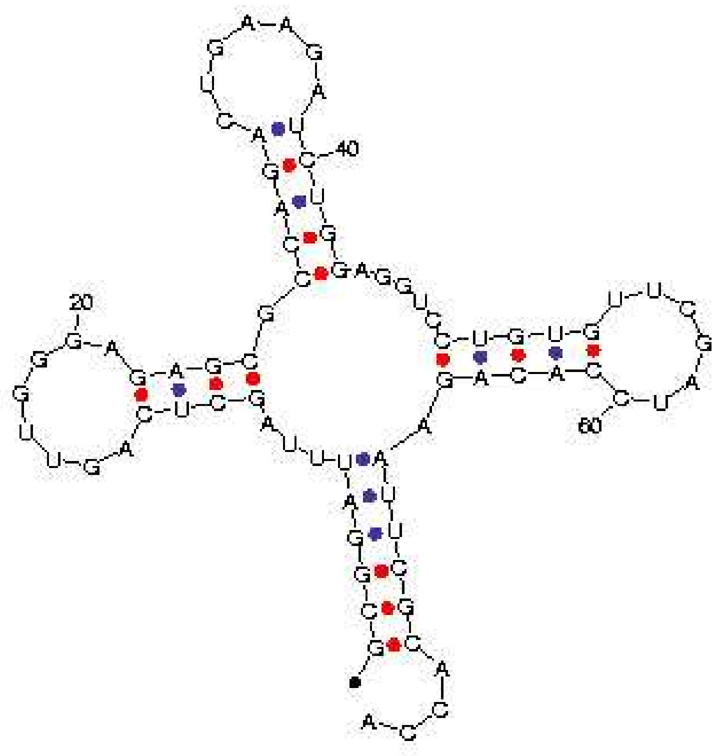
- BPs are important stochastic processes and have been studied since the 19th century.
  [Francis Galton and Rev. H. W. Watson, 1874] used single-type BPs to explain extinction of some family names in the British Peerage.

  Many applications:
  in population dynamics (see, e.g., [Jagers75,....]);
  in nuclear chain reactions: [Everett and Ulam, Los Alamos 1948]

- RMCs also generalize *Stochastic Context-Free Grammars* (SCFGs) which are essentially equivalent to MT-BPs. SCFGs have been studied extensively since the 1970's in Statistical Natural Language Processing (see, e.g., [Manning-Schütze'99]).

  SCFGs have also been used in computational biology for modeling and prediction of secondary structure in tRNA
  (see, e.g., [Sakakibara, Haussler et. al., *Nucleic Acids Research*, 1994]).

tRNA secondary structure and SCFGs

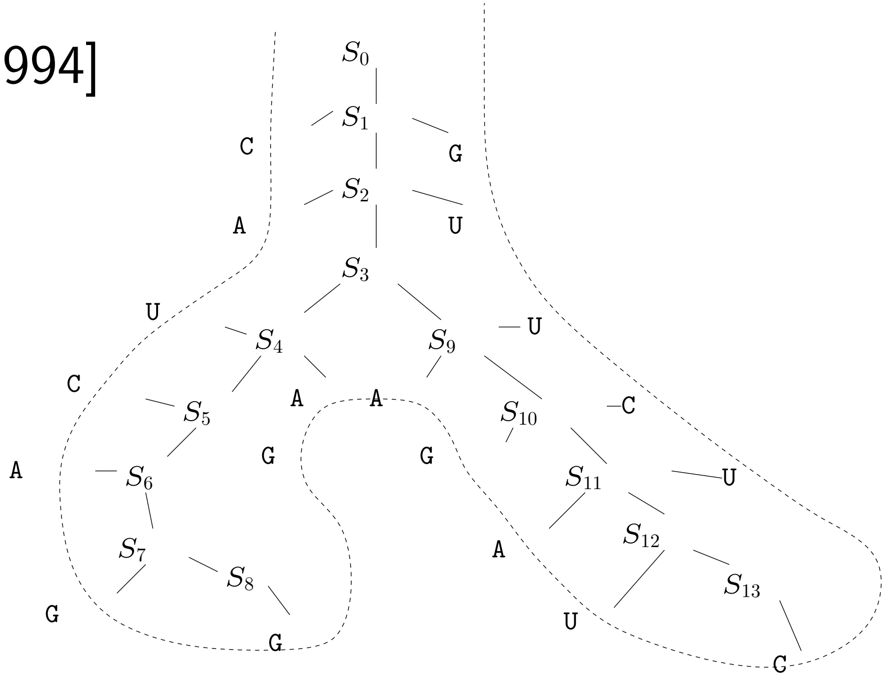


A picture of tRNA secondary (folding) structure (taken from the web).

Modeling of tRNA folding structure using SCFGs

[Sakakibara et. al., 1994]

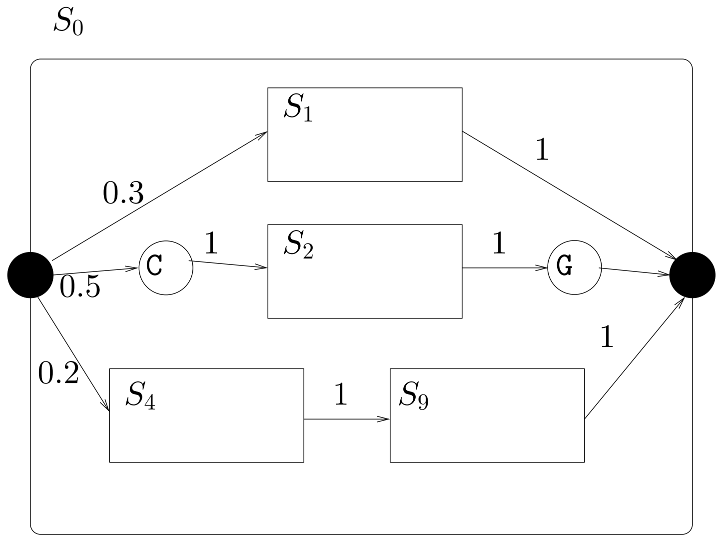
$S_0 \stackrel{0.3}{\rightarrow} S_1$
$S_1 \stackrel{0.5}{\rightarrow} \mathtt{C}S_2\mathtt{G}$
$S_2 \stackrel{0.2}{\rightarrow} \mathtt{A}S_3\mathtt{U}$
$S_3 \stackrel{0.1}{\rightarrow} S_4S_9$
$S_6 \stackrel{1.0}{\rightarrow} \mathtt{A}S_7$
$S_8 \stackrel{0.8}{\rightarrow} \mathtt{G}\ \ldots\ldots$



A parse tree of the (stochastic) context-free grammar describes a *folded* tRNA. The SCFG is statistically learned, via a generalization of HMM algorithms to SCFGs, called the "Inside-Outside" algorithm [Baker'79,Lari-Young,1990].

Why SCFGs (and Multi-Type Branching Processes) are 1-exit RMCs

$$S_0 \xrightarrow{0.3} S_1$$
$$S_0 \xrightarrow{0.5} \mathtt{C} S_2 \mathtt{G}$$
$$S_0 \xrightarrow{0.2} S_4 S_9$$



1-exit RMDPs as "controlled" Branching Processes

Multi-Type Branching Processes (MT-BPs) are the same as a SCFG, except:

1. We don't have terminal symbols.
2. We don't care about the order of nonterminal symbols on the right hand side of rule. (So, the RHS of each rule is just a multi-set of types.)

MT-BPs are basic models in, e.g., mathematical population biology, used for model the stochastic growth of a population of objects of distinct types.

What if there were some types in the population whose "growth" was under our control, while other type exhibit stochastic behavior?
For example, the controlled types could represent those aspects of an ecological system that are under human control.

Could we use "controlled" MT-BPs, i.e., 1-exit RMDPs, to model and analyse whether there is a strategy for us to avoid "almost sure extinction"?

Conclusions

There are many fascinating open questions and directions for future research.
I leave you with just one open question, to whet your appetite:

Question: *Is there a polynomial-time algorithm for answering the quantitative termination problem for 1-exit RSSGs?*

If the answer is yes, it would answer, positively, all of the following:

1. Emerson-Jutla's *Parity Game problem*.
2. Ehrenfeucht-Mycielski's *Mean Payoff Game problem*.
3. Condon's quantitative *Simple Stochastic Game problem.*
4. Garey-Graham-Johnson's *Square-Root Sum problem*.

These are all long standing open problems.