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 a series of papers at STACS’05, TACAS’05, ICALP’05, QEST’05)

Dagstuhl Seminar on Synthesis and Planning, June 12-17, 2005
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 \( \omega \)-regular properties of recursive graphs with the same model complexity.)
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 $\omega$-regular property (given by a Büchi automaton)?
Let’s calculate this termination probability

Let $x$ be the (unknown) probability that starting at $a$ (in the empty calling context) we will eventually reach $b$ (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 (degenerate) 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: badly irrational probabilities

\[
\text{Equation } x = (1/6)x^5 + 1/2. \text{ Thus, } (1/6)x^5 - x + 1/2 = 0.
\]

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: \( x(A_n, en, ex_n') = \frac{1}{2^{2n}} \) and \( x(A_n, en, ex_n'') = 1 - \frac{1}{2^{2n}} \).
Motivation for studying RMCs

- Recursive Graphs and Recursive State Machines (RSMs) ([AlurEtessamiYannakakis’01],[BenediktGodefroidReps’01]) are a natural 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 extensively since the 1970’s in the Natural Language Processing community.

RMCs also generalize Multi-Type Branching Processes (MT-BPs). ([Galton-Watson, 1874], [Kolmogorov-Sevastianov’48], [Everett-Ulam’48], [Harris’63],...). Branching processes are a fundamental class of 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 $\delta_i$, 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 $e x$, both of some component $A_i$ in a RMC, what is the probability of eventually reaching (and terminating at) the global exit state $\langle e, e x \rangle$ of $\mathcal{M}_A$ starting at the global state $\langle e, u \rangle$?

Let us denote this (unknown) probability by $x(i, u, e x)$. (Using these probabilities, we can also calculate other reachability probabilities.)
The non-linear system associated with an RMC

What is \( x(f,z,z) \)?
What is \( x(f,a,z) \)?
What is \( x(f,(b1,c),z) \)?

\[
\begin{align*}
x(f,z,z) & = 1 \\
x(f,a,z) & = \frac{1}{3} x(f,h,z) + \frac{2}{3} x(f,(b1,c),z) \\
x(f,(b1,c),z) & = x(g,c,d) x(f,(b1,d),z) + x(g,c,e) x(f,(b1,e),z)
\end{align*}
\]

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 $x = P(x)$

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

By a Tarski-Knaster argument, $P()$ has a Least Fixed Point $x^*$ 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, $x^* = \lim_{m \to \infty} P^m(0)$, gives precisely the reachability probabilities we are after.

- What can these probabilities be? Can we compute them 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:

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

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,...]), the following is known:
Theorem ([Canny’89, Reneger’92, Basu-et. al.’96, ....]) Suppose we are given such an existential sentence $\varphi \equiv \exists x_1, \ldots, x_k B(\bar{x})$, with $m$ distinct polynomials $f_i$, each of maximum (multi-variate) degree $d$, over variables $x_1, \ldots, x_n$, and with rational coefficients encodable in $L$ bits\(^1\).

There is an algorithm that decides whether $\varphi$ is true when interpreted over the real numbers. The algorithm runs in PSPACE and simultaneously in sequential time:

$$O( (L \log L \log \log L) m (m/n)^n d^{O(n)} )$$

(This is a deep result.)

\(^1\)and where we assume, only for simplicity, that the boolean combination underlying $B(\bar{x})$ can be evaluated in constant time given a truth valuation of the polynomial predicates.
Using the $\exists$-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:

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

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 $\exists$-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 \textit{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. (N.B. Bounded RMCs have no direct syntactic analog in Pushdown systems.)

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

The proof is non-trivial. We use \( \exists \)-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. (I can’t provide details here.)
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 [Sevastianov’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. 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) $0 < x_i^* < 1$. 

Kousha Etessami  RMCs,RMDPs, & RSSGs
Linearly-recursive RMCs

Definition We call an RMC \textit{linearly recursive} (l\textsubscript{r}-RMC) if there is no path of transitions, inside any component, from some box-exit to some box-entry.

(Thanks to Amir Pnueli for asking about l\textsubscript{r}-RMCs at my TACAS’05 talk.)

Theorem \textit{For l\textsubscript{r}-RMCs, there is a P-time algorithm for calculating the exact (rational) termination probabilities, }x\textsubscript{i}^*\text{.}

We will later give a MUCH MORE GENERAL algorithm and result, from which the above theorem follows as a very special case.
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.\(^2\)

---

\(^2\)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.
RMCs and Newton’s method

Recall Newton’s method for univariate polynomial equations $f(x) = 0$: compute the sequence $x_0, x_1, \ldots, x_k, \ldots$, where

$$x_{k+1} := x_k - \frac{f(x_k)}{f'(x_k)}$$

There is a multi-variate generalization of this method, for $F(\bar{x}) = 0$: compute the sequence $x_0, x_1, x_2, \ldots$, where:

$$x_{k+1} := x_k - (F'(x_k))^{-1}F(x_k)$$

Here $F'(x)$, is the *Jacobian matrix*, of partial derivatives and is given by

$$F'(x) = \begin{bmatrix}
\frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\
\vdots & \ddots & \vdots \\
\frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n}
\end{bmatrix}$$
general facts about Newton’s method

- The method won’t even be defined unless the matrix $F'(x_k)$ 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.
Mr. Newton was a very clever man

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

**Theorem 6** Newton's Method, started at 0 on the decomposed system, “monotonically converges” to the LFP. meaning that $(F'(x_k))^{-1}$ exists for all $k \geq 0$, and $0 = x_0 \leq x_1 \leq x_2 \leq \ldots \leq x^*$, and $\lim_{k\to\infty} x_k = x^*$.

Moreover, for all $k \geq 0$, $x_k \geq P^k(0)$.

This is a very useful result in practice. In fact, our proof shows that Newton constitutes a rapid “acceleration” of standard iteration $P^k(0)$. In the known examples where $P^k(0)$ takes exponentially many iterations to achieve $i$ bits, Newton takes linearly many. We **conjecture**: polynomially many iterations in $i$ and $|A|$ are sufficient to achieve $i$ bits (in the unit-cost real RAM model).
Decomposed Newton’s method and linearly-recursive RMCs

Proposition Our decomposed Newton’s method, applied directly to the system \( x = P(x) \) for \( 1 \times \) RMCs computes the exact rational termination probabilities \( x^* \).

More generally, it computes such exact rational values for all piecewise linearly-recursive RMCs, where the strongly-connected components of the (non-linear) system \( x = P(x) \) 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 execution 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 $\omega$-regular model checking

<table>
<thead>
<tr>
<th>Theorem:</th>
<th>reachability</th>
<th>det. Büchi</th>
<th>nondet. Büchi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Qualitative:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1-exit</td>
<td>P</td>
<td>P</td>
<td>P in RMC, EXPTIME in Büchi</td>
</tr>
<tr>
<td>Bounded</td>
<td>P</td>
<td>P</td>
<td>P in RMC, EXPTIME in Büchi</td>
</tr>
<tr>
<td>General</td>
<td>PSPACE</td>
<td>PSPACE</td>
<td>PSPACE in RMC, EXPTIME in Büchi</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>reachability</th>
<th>det. Büchi</th>
<th>nondet. Büchi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quantitative:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1-exit</td>
<td>PSPACE</td>
<td>PSPACE</td>
<td>PSPACE in RMC, EXPSPACE in Büchi</td>
</tr>
<tr>
<td>Bounded</td>
<td>P</td>
<td>P in RMC for fixed Büchi</td>
<td>P in RMC, for fixed Büchi</td>
</tr>
<tr>
<td>General</td>
<td>PSPACE</td>
<td>PSPACE</td>
<td>PSPACE in RMC, EXPSPACE in Büchi</td>
</tr>
</tbody>
</table>

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 [AlurEtessamiYannakakis’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 $\text{ne}(u) = 1 - \sum_{ex \in E_x} x^*_u \cdot x_{(u, ex)}$, be the probability of never exiting the component when starting at $u$. 
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.: $P_A(\square \lozenge v) = \text{probability of reaching bottom SCC containing } v \text{ in } M'_A$. 

Kousha Etessami

RMCs, RMDPs, & RSSGs
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 ([Sevastianov’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-Etessami, 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.
The global infinite SSG

An RSSG $A$ defines a denumerable SSG $M_A = (V, \Delta, \text{player})$. Let $\Psi_i$ 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 $x^{*,\sigma,\tau}_{(u,ex)} = \sup_{\sigma \in \Psi_1} \inf_{\tau \in \Psi_2} x^{*,\sigma,\tau}_{(u,ex)}$.

It follows from general determinacy results (e.g., Martin’s (1998) “Blackwell Determinacy”, which applies more generally to all countable zero-sum stochastic games) that RSSG termination games are **determined**:

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

**Central Algorithmic Problem** Calculate the value $x^{*,\sigma,\tau}_{(u,ex)}$ of the termination game (starting at $u$ and terminating at $ex$).
1-exit RSSGs and nonlinear min/max equations

What is \( x(f,z,z) \)?
What is \( x(f,a,z) \)?
What is \( x(f,(b1,c),z) \)?
What is \( x(f,h,z) \)?
What is \( x(f,(b1,d),z) \)?

We get a new system of the form \( \bar{x} = P(\bar{x}) \).
Basic facts about \( x = P(x) \) (déjà vu)

- The coefficients in \( P() \) are non-negative, and in fact \( P : \mathbb{R}^n \rightarrow \mathbb{R}^n \) defines a monotone operator on a “downward closed”, compact, subspace \( D \) of \([0, 1]^n\).

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

- **Theorem** The LFP, \( x^* = \lim_{m \rightarrow \infty} P^m(0) \), gives precisely the values of the games started at the given vertices.

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

- Can we compute these values efficiently?
  
  You bet!

  **Theorem** We can decide whether \( x^*_{(u, e)} \geq p \), for a given rational \( p \), in PSPACE (again, using the existential theory of reals).
1-exit RSSGs and S&M Determinacy

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

**Definition** (S & M) 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).

Such a 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 extremely delicate, relying on properties of certain power series that arise when studying 1-exit RSSGs.
better results for special subproblems

**Theorem** For 1-exit RMDPs (1-exit RSSGs, respectively) the qualitative termination question, i.e., is $x^*_{(u,ex)} = 1$, can be answered in NP (respectively, in $\Sigma_P^2 \cap \Pi_P^2$).

Proof: Guess S&M strategy, and verify using our P-time algorithm for qualitative termination of 1-exit RMCs (based on an eigenvalue characterization).

**Theorem** For linearly-recursive 1-exit RMDPs (1-exit RSSGs) the exact, rational, value of the game can be computed in polynomial time (in NP $\cap$ co-NP, respectively).

Proof idea: Again a decomposition into linear subproblems.
Ok, what about RMDPs with 2 exits?

![Diagram of a 2-exit RMDP]

**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 at all!

The strategy $L^nR$ 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 $\epsilon$-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 
\[ x^*(u,ex) = 1. \]  This is so 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 
\[ x^*(u,ex) \text{ is } 1 \text{ or is } < \epsilon. \]

We prove this via a reduction from the emptiness problem for *Probabilistic Finite Automata* (PFA) [Paz’71, Condon-Lipton’89]. We use the more recently proved fact ([Blondel-Canterini’2003]) that the PFA emptiness problem is undecidable even when the number of states is bounded by a fixed constants.
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-Strazovský’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 \(\omega\)-regular model checking, 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, to appear] we extended our study of RMCs to RMDPs and RSSGs.

• In [Yannakakis-E., QEST’05, to appear], we extended our model checking study to the case of LTL specifications.
Conclusions

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

**Question:** *Is there a polynomial-time algorithm for answering the quantitative termination question 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. Condon’s *Simple Stochastic Game problem*.

Each of these is a FUNDAMENTAL open problem.

(Currently, for the *Square-Root Sum problem*, the best we know is essentially containment in PSPACE, but there are conjectures in Diophantine approximation theory which, if true, would yield a P-time algorithm for Square-Root Sum.)