Algorithms for Linear Equations with Min and Max Operators Under (Absolutely) Halting Condition

Krishnendu Chatterjee
IST Austria
Klosterneuburg, Austria
kchatterjee@ist.ac.at

Ruichen Luo
IST Austria
Klosterneuburg, Austria
rluo@ist.ac.at

Raimundo Saona
London School of Economics
London, United Kingdom
raimundo.saona@gmail.com

Jakub Svoboda
IST Austria
Klosterneuburg, Austria
jakub.svoboda@ist.ac.at

Abstract

We consider linear equations with min and max operators (LEMMs) that contains many subproblems ranging from optimization, games, to model checking. Recently, Chatterjee et al. [4] give a systematic study of the complexity of different subclasses. Three key subclasses—(i) halting branching process, (ii) absolutely halting LEMMs, and (iii) halting LEMMs—are identified as being in UP ∩ coUP while generalizing stochastic games. In this work, we study the classic algorithms of Policy Iteration (PI) and Value Iteration (VI) for these general subclasses.

First, we simplify the problem hierarchy by showing the equivalence between halting branching process and stochastic games. Then, we show that PI diverges for absolutely halting LEMMs, while VI converges following from a new analysis based on spectral gap. Applying our new analysis back to well-studied subproblems of stochastic games yields surprising improvements: we refine the long-standing analyses of VI and PI for reachability objectives using the spectral gap, and improve the best-known strongly polynomial rate of PI for discounted-sum objectives (with fixed discount factor) by a logarithmic factor. Finally, we show that neither PI nor VI converges for general halting LEMMs and, to this end, propose variants of simple policy iteration that ensure convergence across all subclasses.

Keywords

Stochastic Games, Policy Iteration, Value Iteration

ACM Reference Format:

Krishnendu Chatterjee, Ruichen Luo, Raimundo Saona, and Jakub Svoboda. 2026. Algorithms for Linear Equations with Min and Max Operators Under (Absolutely) Halting Condition. In . ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Conference'17, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

Optimization problem. Optimization is in the core of many problems in formal methods, programming languages, logics, and artificial intelligence. Prominent examples include model checking, probabilistic program analysis, constraint programming, reinforcement learning, evolutionary games, to name a few. In this work, we are interested in the optimization problem of solving for $\mathbf{x} = [x_1, \ldots, x_n]^\top \in \mathbb{R}^n$ in the system of *Linear Equations with Min and Max operators* (LEMM) given by $(S_{\min}, S_{\max}, S_{\text{aff}}, n, \mathcal{N}, \mathbf{q}, \mathbf{b})$:

$$\begin{cases} x_i = \min_{l \in \mathcal{N}(i)} x_l, & i \in S_{\min}, \\ x_j = \max_{l \in \mathcal{N}(j)} x_l, & j \in S_{\max}, \\ x_k = \mathbf{q}_k^\top \mathbf{x} + b_k, & k \in S_{\text{aff}}, \end{cases} \tag{1}$$

where the following conditions are satisfied: (a) $S_{\min}$, $S_{\max}$, and S_{aff} are disjoint sets such that $S_{\min} \cup S_{\max} \cup S_{\text{aff}} = [n]$,[1] (b) $\emptyset \subsetneq \mathcal{N}(i) \subseteq [n]$ for $i \in S_{\min} \cup S_{\max}$, (c) $\mathbf{q}_k \in \mathbb{R}^n$ for $k \in S_{\text{aff}}$, and (d) $\mathbf{b} = [b_1, \cdots, b_n]^\top \in \mathbb{R}^n$ such that $b_i = 0$ for $i \in S_{\min} \cup S_{\max}$.

It is known from the literautre [4] that Problem (1) covers many applications, such as linear program with boolean variables [17], verifying neural networks [9], constraint satisfaction problems [2], and evolutionary in ecosystems [19].

Restrictive conditions. Given the generality and hardness of the problem, researchers focus on several natural subclasses and let us first recall the restrictive conditions: (C1) the operators are halting; (C1+) the operators are absolutely halting; (C2) the coefficients are non-negative; (C3) the rows sum up to one; and (C4) there is only min operator or only max operator. Various subsets of conditions yield various subproblems. For instance, Condition C2 models branching processes; Conditions C2 and C3 together model probabilistic transitions; and Condition C1 or Condition C1+ implies that the underlying state transitions satisfies stability or absolute stability.

Well-studied subproblems. A few well-studied problems correspond to different subclasses: with min and max operators, under Conditions C2 and C3 we obtain (two-player turn-based zero-sum) *stochastic games* with reachability objectives; the seminal paper [7] shows that the halting condition C1 can be further assumed without loss of generality, which leads to halting stochastic games (a.k.a.

[1] Denote $[k] := \{1, 2, \cdots, k\}$, for all $k \in \mathbb{Z}_{\geq 1}$.

simple stochastic games or SSGs); while when there is only one type of opearator—that is, under Condition C4—the stochastic game is reduced to Markov decision process (MDP) with reachability objectives.

Key open subproblems. Recently, Chatterjee et al. [4] give a systematic study of the computational complexities of the LEMM subproblems under all subsets of conditions. All the subproblems are classified into the NP-complete category, the UP ∩ coUP (while no easier than SSGs) category, and the category solvable in polynomial time. In this paper, we study algorithms for LEMM subproblems, and in particular, for the ones in the second category—UP ∩ coUP—because from an algorithmic perspective, the second UP ∩ coUP category represents the most exciting challenges. The subproblems in this category are not NP-hard (unless NP = coNP), yet generalize the fundamental and difficult problem of SSGs, for which the existence of polynomial time algorithms is a major open problem in the field.

Policy Iteration and Value Iteration. Since the key open subproblems are natural generalization of SSGs, the classic algorithms for SSGs are the natural candidates for these subproblems. The two most fundamental algorithms for SSGs are Policy Iteration (PI) and Value Iteration (VI) [1, 6]: PI iteratively updates the policy of one player based on the best response of the other player; VI iteratively updates the value vector by propagating the last value vector through the games. In this work, we consider natural extensions of both PI and VI to the framework of LEMMs, and investigate their convergence behaviors for the different general subclasses.

Related work on the runtime analyses of PI *and* VI*.* It is practically observed that both PI and VI converge fast for most instances in SSGs and MDPs, but both algorithms take exponential time in the worst case. The special class of discounted-sum objectives is relatively well understood in theory: when the discount factor γ is bounded away from 1, VI converges in polynomial time [11], and PI converges in strongly polynomial time due to the seminal result of [8, 20]. However, for the (more general) halting MDPs and stochastic games with reachability objectives, the runtime analyses are considerably more elusive. The only known upper bound for PI in general is the exponential bound from enumerating all the possible policies. The currently best upper bound for VI is a convergence rate of $\tilde{O}(\frac{1}{1-\tilde{\gamma}})$,[2] where $(1-\tilde{\gamma})$ describes the guaranteed probability of reaching sink states after every n steps. Therefore, the following problems remain open:

- *Whether the dependence on $(1-\tilde{\gamma})$ in the current analysis of* VI *for MDPs and SSGs can be improved*?
- *whether the (strongly) polynomial runtime results for discounted-sum objectives can be generalized to broader classes with reachability objectives*?
- *Whether the* PI *and* VI *still converge for various LEMM generalizations?*

Related work on Markov chain and spectral gap. Markov chain is the fundamental model underpinning the state transitions in a probabilistic system. Its *mixing time* measures how quickly the system evolves into a steady state distribution. An important tool from spectral analysis is the *spectral gap*—the difference between the transition matrix's two largest eigenvalues[3]—that can be used to control the mixing time [10]. However, to our knowledge, this concept of spectral gap has never been generalized to the problems of MDPs and stochastic games with reachability objectives. Therefore, the following question remain open:

- *Whether one can use the tool from spectral analysis to explore the connection between spectral gap and the convergence of* VI *and* PI *for MDPs, SSGs, or even halting LEMMs?*

Our contributions. In this work, we give a systematic algorithmic study on the convergence of PI and VI for various LEMM subproblems in UP ∩ coUP. Moreover, when applying our new approach back to the fundamental problem of SSGs, we have (surprisingly) improved the long-standing running time analyses of VI and PI, which is a key result of this work. In detail, our contributions are listed as follows:

(1) We show the equivalence between halting branching process and SSG by a linear reduction (Theorem 1).
(2) We consider absolutely halting LEMMs and show that PI diverges yet VI converges (Theorem 2). The key technique of our VI analysis is a new preconditioning that connects the convergence rate and the spectral gap (Lemmas 2 and 3).
(3) We consider our new VI analysis approach for the fundamental problems of MDPs and stochastic games:
(3a) For halting MDPs and stochastic games with reachability objectives, by connecting the convergence rates of VI and PI with spectral gap, we refine the long-standing bound of VI (Corollary 3) and obtain a new bound of PI (Theorem 4);
(3b) For stochastic games with discounted-sum objectives, we improve the best-known strongly polynomial time bound of PI by a logarithmic factor (Corollary 5).
(4) We consider general halting LEMMs and show that neither PI nor VI converges. To this end, we revisit SPI and RandSPI and prove their convergence for the general class (Theorem 6).

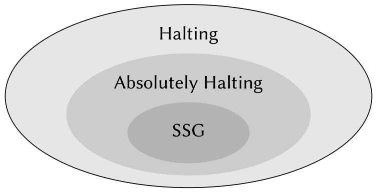
2 Preliminaries

LEMM and its decision problem. We refer to Eq. (1) as the LEMM with $(S_{\min}, S_{\max}, S_{\text{aff}}, n, \mathcal{N}, \mathbf{q}, \mathbf{b})$. If $\mathbf{x} \in \mathbb{R}^n$ satisfies Eq. (1), we say that $\mathbf{x}$ is a *(feasible) solution* to the LEMM. We define the *LEMM decision problem* as follows: given an LEMM with $(S_{\min}, S_{\max}, S_{\text{aff}}, n, \mathcal{N}, \mathbf{q}, \mathbf{b})$, a threshold $\beta \in \mathbb{R}$, and an index $i \in [n]$, decide whether there exists a feasible solution $\mathbf{x}$ of the given LEMM, such that $x_i < \beta$.

Vectors, matrices and sets. Let $\mathbf{e}_i = [\delta_{i,1}, \ldots, \delta_{i,n}]^\top$, $i \in [n]$, where $\delta_{i,j}$ is 1 if $i = j$ and 0 otherwise. Let $\mathbf{0}_k \in \mathbb{R}^k$ (or $\mathbf{1}_k \in \mathbb{R}^k$) denote the vector where every element in the vector equals 0 (or 1). For any vector $\mathbf{v} \in \mathbb{R}^n$, let $\text{diag}(\mathbf{v}) \in \mathbb{R}^{n\times n}$ denote the diagonal matrix with $\mathbf{v}$ on its diagonal. Let $\mathbf{O}_{k_1\times k_2} \in \mathbb{R}^{k_1\times k_2}$ denote the matrix where every element equals 0. Let $\mathbf{I}_k \in \mathbb{R}^{k\times k}$ denote the identity matrix. The subscripts k, k_1, and k_2 might be obsolete when they are clear from the context. For any matrix $\mathbf{Q} \in \mathbb{R}^{n\times n}$, let $\varrho(\mathbf{Q})$ denote its spectral radius; let $\det(\cdot)$ denote its determinant; let $|\mathbf{Q}| \in \mathbb{R}^{n\times n}$ denote the matrix where each element is the absolute value of the

[2]In the $\tilde{O}(\cdot)$ notation, the polynomial terms are obsolete.

[3]The largest eigenvalue in a Markov chain is trivially 1. Let γ be the second largest eigenvalue, and the spectral gap is given by $(1-\gamma)$.



Problem Classes	Checking	PI	VI	SPI
{C1}, {C1,C3}, {C1,C4}, {C1,C3,C4}	coNP-*comp.*	✘	✘	✔
{C1+}, {C1+,C3}, {C1+,C4}, {C1+,C3,C4}	PTIME	✘	✔	✔
{C1,C2}, {C2,C3}, {C1,C2,C3}	PTIME	✓	✓	✓

Figure 1: *Three key categories in the complexity class UP ∩ coUP while no easier than SSGs: the subproblems, the complexity of checking the conditions, and the convergence of* PI, VI *and* SPI *algorithms. The rows of the subproblems are ordered by decreasing generality (and thus difficulty) from top to bottom. The subproblems in the same row are linearly equivalent to each other. The new results from this paper are in* **bold and red**.

corresponding element in $\mathbf{Q}$; and let $\mathbf{Q}_{-ij} \in \mathbb{R}^{(n-1)\times(n-1)}$ denote the matrix formed from $\mathbf{Q}$ by deleting its ith row and jth column. For any finite set, let $|\cdot|$ denote the number of elements in the set.

Notations. Consider the LEMM with ($S_{\min}$, $S_{\max}$, S_{aff}, n, $\mathcal{N}$, $\mathbf{q}$, $\mathbf{b}$). Let $n_1 = |S_{\min}|$ and $n_2 = |S_{\max}|$. Let $m = \sum_{i\in S_{\min}\cup S_{\max}} |\mathcal{N}(i)|$. Without loss of generality, we assume $S_{\min} = \{1, \cdots, n_1\}$ and $S_{\max} = \{n_1+1, \cdots, n_1+n_2\}$, and then, we denote

$$\mathcal{Q}_{\min} = \left\{ [\mathbf{e}_{\ell_1}, \cdots, \mathbf{e}_{\ell_{n_1}}]^\top \,\middle|\, \ell_i \in \mathcal{N}(i) \text{ for } i \in S_{\min} \right\},$$
$$\mathcal{Q}_{\max} = \left\{ [\mathbf{e}_{\ell_{n_1+1}}, \cdots, \mathbf{e}_{\ell_{n_1+n_2}}]^\top \,\middle|\, \ell_j \in \mathcal{N}(j) \text{ for } j \in S_{\max} \right\},$$
$$\mathbf{Q}_{\text{aff}} = [\mathbf{q}_{n_1+n_2+1}, \ldots, \mathbf{q}_n]^\top .$$

Let

$$\mathcal{Q} = \left\{ \begin{bmatrix} \mathbf{Q}_{\min} \\ \mathbf{Q}_{\max} \\ \mathbf{Q}_{\text{aff}} \end{bmatrix} \,\middle|\, \mathbf{Q}_{\min} \in \mathcal{Q}_{\min} \text{ and } \mathbf{Q}_{\max} \in \mathcal{Q}_{\max} \right\},$$

and let $\text{conv}(\mathcal{Q})$ denote the convex hull of $\mathcal{Q}$. When all input numbers are rational, for all $k \in S_{\text{aff}}$, let $d_k \in \mathbb{Z}_{\geq 1}$ be the least common multiple of $\mathbf{q}_k$ and b_k (that is, $d_k\mathbf{q}_k \in \mathbb{Z}^n$ and $d_k b_k \in \mathbb{Z}$). Denote

$$D = \max_{k\in S_{\text{aff}}} d_k \text{ and } \varepsilon^* = 2^{-(n+1)} D^{-n}. \tag{2}$$

We say that $\mathbf{x} \in \mathbb{R}^n$ is an *ε^*-accurate solution*, if there exists a solution $\mathbf{x}^*$ such that $\|\mathbf{x} - \mathbf{x}^*\|_\infty \leq \varepsilon^*$.

2.1 Restrictive conditions

The LEMM decision problem is NP-complete in general, while to obtain computationally tractable problem classes several restrictive conditions are introduced in the literature [4, 7, 14].

Condition C1 (Halting). *For all* $\mathbf{Q} \in \text{conv}(\mathcal{Q})$*, we have that* $\lim_{k\to\infty} \mathbf{Q}^k = \mathbf{O}_n$.

Condition C1+ (Absolutely halting). *For all* $\mathbf{Q} \in \mathcal{Q}$*, we have that* $\lim_{k\to\infty} |\mathbf{Q}|^k = \mathbf{O}_n$.

Condition C2 (Non-negativity). *For all* $k \in S_{\text{aff}}$*, we have that* $\mathbf{q}_k \geq 0$ *and* $b_k \geq 0$.[4]

Condition C3 (Sum upto one). *For all* $k \in S_{\text{aff}}$*, we have that* $\mathbf{q}_k \geq 0$*,* $b_k \geq 0$*, and* $\mathbf{q}_k^\top \mathbf{1} + b_k \leq 1$.

Condition C4 (Max-only or min-only). *Either* $S_{\min} = \varnothing$ *or* $S_{\max} = \varnothing$.

[4]Throughout this paper, all equalities, inequalities, and min/max operations involving two vectors, or a vector and a real number, are understood element-wise; that is, they are applied separately to each corresponding coordinate.

We use the following subset notation for the LEMM decision problem under various subsets of the conditions. For instance, "the LEMM decision problem under {C1,C2}" means "the LEMM decision problem under Conditions C1 and C2".

We also remark that, while Condition C1+ is in general stronger than Condition C1, under Condition C2 they become trivially equivalent. In other words, the set of conditions {C1+,C2} is equivalent to the set of conditions {C1,C2}.

Discussion about related problem classes. The conditions considered above are natural in relevant literature. The LEMM under {C2,C3,C4} corresponds to *Markov decision process* (MDP) with reachability objectives; the LEMM under {C2,C3} corresponds to (two-player turn-based zero-sum) *stochastic games* with reachability objectives; the LEMM under {C2} corresponds to *branching process*. In the contexts of MDP, stochastic games, and branching process, the Condition C1 is known as halting condition characterizing the stability of the underlying state transitions. For instance, the halting stochastic games (a.k.a. simple stochastic games or SSGs) correspond to the LEMM under {C1,C2,C3}.

2.2 Computational complexity results from literature

Complexity of the LEMM decision problem. The MDP (or the LEMM under {C2,C3,C4}) is known to be solvable in polynomial time [14]. The stochastic games (or the LEMM under {C2,C3}) are known in the complexity class of UP ∩ coUP [3, 7], yet the existence of a polynomial-time algorithm remains a major open problem. Condon [7] shows that stochastic games are polynomially equivalent to SSGs (or the LEMM under {C1,C2,C3}).

More recently, Chatterjee et al. [4] give a systematic study of the computational complexities under different subsets of conditions. In detail, they show that the LEMM under {C2,C4} or {C3,C4} is NP-hard; while the halting LEMM (that is, under {C1}) has a unique solution and belongs to the complexity class UP ∩ coUP; and moreover, the LEMM under {C1,C3}, {C1,C4}, or {C1,C3,C4} is no easier than the halting LEMM. In particular, they indentify all the subproblems between the halting LEMMs and the SSGs as belonging to the complexity class UP ∩ UP yet no easier than SSGs.

Complexity of checking the conditions. The complexity of checking the subsets of conditions has also been studied in the literature [4, 14]. While Conditions C2 to C3 can be easily checked in linear time, [4] further show that Condition C1+ can be checked in polynomial time yet checking Condition C1 is coNP-hard. Together with the complexity of the LEMM decision problem, this implies that the language of "LEMM decision problems under Condition C1+ with 'yes' answers" is in UP ∩ coUP.

We summarize the complexity results known from the literature.

Proposition 1 ([4, 7, 14]). *The following assertions hold:*

- *The LEMM decision problem under* {C2,C4} *or* {C3,C4} *is NP-complete.*
- *The LEMM under* {C1} *has a unique solution, and thus, its associated decision problem is in UP ∩ coUP.*
- *The LEMM under* {C2,C3,C4} *or* {C1,C2,C4} *is polynomially solvable, and thus, their associated decision problems are in PTIME.*
- *The LEMMs under* {C1} *and the LEMMs under* {C1,C3,C4} *are linearly equivalent.*
- *The LEMMs under* {C1,C2,C3}*, the LEMMs under* {C2,C3}*, and the SSGs are linearly equivalent.*
- *The LEMMs under* {C1,C2,C3,C4}*, the LEMMs under* {C2,C3,C4}*, and the MDPs with reachability objectives are linearly equivalent.*
- *Checking Condition C1 is coNP-complete;*[5] *checking Condition C1+ is in PTIME; and checking Conditions C2 to C4 can be done in linear time.*

Remark 1. *In Proposition 1, we know from the literature that all (except two) of the LEMM subproblems are classified precisely into three complexity categories: NP-complete, UP ∩ coUP (while no easier than SSGs), and PTIME. In detail:*

- *NP-complete:* {C2,C4}, {C3,C4}, {C2}, {C3}, {C4}, *and* ∅*;*
- *UP ∩ coUP (while no easier than SSGs):* {C1}, {C1,C3}, {C1,C4}, {C1,C3,C4}, ***{C1+}, {C1+,C3}, {C1,C2}***, {C2,C3}, *and* {C1,C2,C3}*;*
- *PTIME:* {C1,C2,C3,C4}, {C2,C3,C4}, *and* {C1,C2,C4}.

There are two remaining subproblems ***{C1+,C4}*** *and* ***{C1+,C3,C4}****, which we do not know from the literature whether they are easier or harder than SSGs. Therefore, these may belong to the UP ∩ coUP category or the PTIME category, and their memberships are not entirely clear from the results in the literature.*

Proposition 1 further classifies the nine of the LEMM subproblems in UP ∩ coUP (while no easier than SSGs) into three key subclasses:

- *Halting LEMMs: the first four LEMM subproblems (*{C1}, {C1,C3}, {C1,C4}, *and* {C1,C3,C4}*) in the top row are linearly equivalent;*
- *Absolutely halting LEMMs: the three LEMM subproblems* ***{C1+}, {C1+,C3},*** *and* ***{C1,C2}*** *are no harder than the first four subproblems in the top row while no easier than the last two subproblems in the bottom row; and*
- *SSGs: the last two LEMM subproblems (*{C2,C3} *and* {C1,C2,C3}*) from the bottom row are linearly equivalent to SSGs.*

For the algorithmic study in this work, we focus on developing algorithms for the subproblems in the (arguably most interesting) complexity class of UP ∩ coUP (while no easier than SSGs). Moreover, we also try to identify the exact memberships for the two remaining subproblems ***{C1+,C4}*** *and* ***{C1+,C3,C4}****.*

[5]In the original paper of [4], checking Condition C1 is proven to be coNP-hard. This complexity result can indeed be strengthened to coNP-completeness. The detailed proof can be found in Section A.

2.3 Classic algorithms: policy iteration and value iteration

As discussed above, [4] highlights a few new LEMM subclasses in UP ∩ coUP while no easier than SSGs. In this paper, we study the algorithms for these subclasses. In this section, we formalize the classic algorithms of Policy Iteration (PI) and Value Iteration (VI), and then review their classic analyses in the literature under the setting of SSGs.

Policy Iteration. Given an LEMM with ($S_{\min}$, $S_{\max}$, S_{aff}, n, $\mathcal{N}$, $\mathbf{q}$, $\mathbf{b}$), for every max policy $\mathbf{Q}_{\max} \in \mathcal{Q}_{\max}$, we define its value estimate $\mathbf{x}_{\max}(\mathbf{Q}_{\max}) \in \mathbb{R}^n$ as the (unique) solution of the following system of equations:

$$\begin{cases} x_i = \min_{l \in \mathcal{N}(i)} x_l, & 1 \le i \le n_1, \\ [x_{n_1+1}, \cdots, x_{n_1+n_2}]^\top = \mathbf{Q}_{\max}\mathbf{x}, & \\ x_k = \mathbf{q}_k^\top \mathbf{x} + b_k, & n_1 + n_2 < k \le n. \end{cases} \tag{3}$$

Based on this value estimate, the max policy extraction $\pi_{\max}\colon \mathbb{R}^n \to \mathcal{Q}_{\max}$ satisfies the following condition: for every $\mathbf{x} \in \mathbb{R}^n$, let $\pi_{\max}(\mathbf{x}) \in \mathcal{Q}_{\max}$ such that

$$\pi_{\max}(\mathbf{x}) \cdot \mathbf{x} = \max_{\mathbf{Q}'_{\max} \in \mathcal{Q}_{\max}} \mathbf{Q}'_{\max} \cdot \mathbf{x}\,.$$

The pseudocode of Policy Iteration algorithm updating max policies (PImax) is given in Algorithm 1. Starting from an initial max policy $\mathbf{Q}^{(0)}_{\max}$, PImax iteratively estimates the value and extracts the next max policy.

Algorithm 1 Policy Iteration updating max policies PImax($\mathbf{Q}^{(0)}_{\max}$)

Require: $\mathbf{Q}^{(0)}_{\max} \in \mathcal{Q}_{\max}$
 for $t = 1, 2, \cdots$ **do**
  $\underline{\mathbf{x}}^{(t-1)} = \mathbf{x}_{\max}(\mathbf{Q}^{(t-1)}_{\max})$
  $\mathbf{Q}^{(t)}_{\max} = \pi_{\max}(\underline{\mathbf{x}}^{(t-1)})$
 end for

Similarly, we can also define value estimate $\mathbf{x}_{\min}\colon \mathcal{Q}_{\min} \to \mathbb{R}^n$, min policy extraction $\pi_{\min}\colon \mathbb{R}^n \to \mathcal{Q}_{\min}$, and have a Policy Iteration algorithm updating min policies (PImin).

It should be noted, however, that greedily updating both min and max policies simultaneously does not converge for SSGs [6], so we only consider the policy iteration updating max or min policies, but not both.

Value Iteration. Given an LEMM with ($S_{\min}$, $S_{\max}$, S_{aff}, n, $\mathcal{N}$, $\mathbf{q}$, $\mathbf{b}$), define the value iteration operator $v\colon \mathbb{R}^n \to \mathbb{R}^n$ as follows: for any $\mathbf{x} \in \mathbb{R}^n$, let $\mathbf{x}' = v(\mathbf{x})$ such that

$$\begin{cases} x'_i = \min_{l \in \mathcal{N}(i)} x_l, & 1 \le i \le n_1, \\ x'_j = \max_{l \in \mathcal{N}(j)} x_l, & n_1 < j \le n_1 + n_2, \\ x'_k = \mathbf{q}_k^\top \mathbf{x} + b_k, & n_1 + n_2 < k \le n. \end{cases} \tag{4}$$

The pseudocode of value iteration algorithm (VI) is given in Algorithm 2. Starting from an initial value vector $\mathbf{x}^{(0)}$, VI iteratively computes the next value vector by the v operator.

Suppose all input numbers are rationals. Let us recall the ε^* defined in Eq. (2). As shown in Proposition 2, while VI itself is

Algorithm 2 Value Iteration $\mathsf{VI}(\mathbf{x}^{(0)})$

Require: $\mathbf{x}^{(0)} \in \mathbb{R}^n$
 for $t = 1, 2, \cdots$ **do**
  $\mathbf{x}^{(t)} = v(\mathbf{x}^{(t-1)})$
 end for

an approximation algorithm, the exact solution can be obtained in polynomial time from any ε^*-accurate solution (provided the halting condition holds). Therefore, the analyses of VI algorithm for LEMMs under halting condition focus on the complexity of reaching an ε^*-accurate solution.

Proposition 2. *Consider an LEMM with* ($S_{\min}$, $S_{\max}$, S_{aff}, n, $\mathcal{N}$, $\mathbf{q}$, $\mathbf{b}$) *under Condition C1. Let* $\mathbf{x}^*$ *be its unique solution. Furthermore, suppose all input numbers are rationals. Let* $\mathbf{x} \in \mathbb{R}^n$ *be an* ε^**-accurate solution, and let* $\mathbf{Q} \in \mathcal{Q}$ *such that* $\mathbf{Qx} + \mathbf{b} = v(\mathbf{x})$*. Then,*

$$\mathbf{x}^* = (\mathbf{I} - \mathbf{Q})^{-1}\mathbf{b} .$$

Proof. By Eq. (2), let

$$\Gamma = \operatorname{diag}(1, \cdots, 1, d_{n_1+n_2+1}, \cdots, d_n) \in \mathbb{Z}^{n\times n}.$$

We have $\Gamma\mathbf{b} \in \mathbb{Z}^n$, and $\Gamma\mathbf{Q} \in \mathbb{Z}^{n\times n}$ for all $\mathbf{Q} \in \mathcal{Q}$.

Let $\mathbf{Q}^* \in \mathcal{Q}$ such that $\mathbf{x}^* = \mathbf{Q}^*\mathbf{x}^* + \mathbf{b}$. Under Condition C1, $(\mathbf{I} - \mathbf{Q}^*)$ is invertible, and we have

$$\mathbf{x}^* = (\mathbf{I} - \mathbf{Q}^*)^{-1}\mathbf{b} = [\Gamma(\mathbf{I} - \mathbf{Q}^*)]^{-1}(\Gamma\mathbf{b}) = \frac{\mathbf{M}}{\det(\Gamma(\mathbf{I} - \mathbf{Q}^*))}(\Gamma\mathbf{b}) ,$$

where $\mathbf{M}$ is the adjugate matrix of $\Gamma(\mathbf{I} - \mathbf{Q}^*)$. In view of $\Gamma\mathbf{b} \in \mathbb{Z}^n$ and $\Gamma\mathbf{Q}^* \in \mathbb{Z}^{n\times n}$, we have $\mathbf{M} \in \mathbb{Z}^{n\times n}$, and then

$$\det(\Gamma(\mathbf{I} - \mathbf{Q}^*))\mathbf{x}^* = \mathbf{M}(\Gamma\mathbf{b}) \in \mathbb{Z}^n.$$

Moreover, we have

$$\det(\Gamma(\mathbf{I}-\mathbf{Q}^*)) \le D^{|S_{\text{aff}}|} \det(\mathbf{I}-\mathbf{Q}^*) \le D^{|S_{\text{aff}}|}(\varrho(\mathbf{I}-\mathbf{Q}^*))^n < D^{|S_{\text{aff}}|}2^n,$$

where the last inequality follows from Condition C1. Therefore, for all $1 \le i, j \le n$,

$$\text{either } x_i^* = x_j^* \text{ or } |x_i^* - x_j^*| \ge \frac{1}{\det(\Gamma(\mathbf{I} - \mathbf{Q}^*))} > (2D)^{-n}.$$

Since $\|\mathbf{x} - \mathbf{x}^*\|_\infty \le \varepsilon^*$, we have for all $1 \le i, j \le n$, if $x_i \ge x_j$, then

$$x_i^* \ge x_i - \varepsilon^* \ge x_j - \varepsilon^* \ge x_j^* - 2\varepsilon^* = x_j^* - (2D)^{-n},$$

and hence, $x_i^* \ge x_j^*$. Therefore, the neighbor choices of $\mathbf{Q}$ in $v(\mathbf{x}) = \mathbf{Qx} + \mathbf{b}$ are made correctly, and $\mathbf{x}^* = \mathbf{Qx}^* + b = (\mathbf{I} - \mathbf{Q})^{-1}\mathbf{b}$. □

Now, we cite the classic analyses of PI and VI algorithms for SSGs in the literature.

Proposition 3 (Algorithm analyses for SSGs [5, 6]). *Consider an LEMM with* ($S_{\min}, S_{\max}, S_{\text{aff}}, n, \mathcal{N}, \mathbf{q}, \mathbf{b}$) *under conditions* {C1,C2,C3}*. Let* $\mathbf{x}^*$ *be its unique solution.*

(1) Let $\mathbf{Q}_{\max}^{(0)} \in \mathcal{Q}_{\max}$ *and* $\mathbf{Q}_{\min}^{(0)} \in \mathcal{Q}_{\min}$ *be any given policies. The following assertions hold for the iterates* $(\underline{\mathbf{x}}^{(t)})_{t\in\mathbb{Z}_{\ge 0}}$ *in* $\mathsf{PImax}(\mathbf{Q}_{\max}^{(0)})$ *and the iterates* $(\overline{\mathbf{x}}^{(t)})_{t\in\mathbb{Z}_{\ge 0}}$ *in* $\mathsf{PImin}(\mathbf{Q}_{\min}^{(0)})$ *:*
 (a) There exists $t \le \Pi_{j\in S_{\max}} |\mathcal{N}(j)| - 1$ *such that* $\underline{\mathbf{x}}^{(t)} = \mathbf{x}^*$*;*
 (b) There exists $t \le \Pi_{j\in S_{\min}} |\mathcal{N}(j)| - 1$ *such that* $\overline{\mathbf{x}}^{(t)} = \mathbf{x}^*$*.*

(2) Let

$$\tilde{\gamma} \triangleq \max \left\{ \|\mathbf{Q}^{(n)} \cdots \mathbf{Q}^{(1)}\|_\infty^{\frac{1}{n}} \,\middle|\, \mathbf{Q}^{(i)} \in \mathcal{Q} \text{ for all } i \in [n] \right\} < 1 . \quad (5)$$

Let $\underline{\mathbf{x}}^{(0)} \in \mathbb{R}^n$ *be any given point such that* $\underline{\mathbf{x}}^{(0)} \le \mathbf{x}^*$*. The following assertions hold for the iterates* $(\underline{\mathbf{x}}^{(t)})_{t\in\mathbb{Z}_{\ge 0}}$ *in* $\mathsf{VI}(\underline{\mathbf{x}}^{(0)})$*:*
 (a) For all $k \in \mathbb{Z}_{\ge 0}$,

$$\max_{i\in[n]} (\mathbf{x}_i^* - \underline{\mathbf{x}}_i^{(kn)}) \le \tilde{\gamma}^n \max_{i\in[n]} \left(\mathbf{x}_i^* - \underline{\mathbf{x}}_i^{((k-1)n)}\right) \le \cdots \le \tilde{\gamma}^{kn} \max_{i\in[n]} \left(\mathbf{x}_i^* - \underline{\mathbf{x}}_i^{(0)}\right) ;$$

 (b) Suppose all input numbers are rationals. There exists

$$t < \frac{1}{1 - \tilde{\gamma}} \left[n \log(2D) + \log\left(2 \max_{i\in[n]} (\mathbf{x}_i^* - \underline{\mathbf{x}}_i^{(0)})\right) \right] + n \quad (6)$$

 such that $\underline{\mathbf{x}}^{(t)}$ *is an* ε^**-accurate solution.*

Similar assertions hold for $\mathsf{VI}(\overline{\mathbf{x}}^{(0)})$ *iterates* $(\overline{\mathbf{x}}^{(t)})_{t\in\mathbb{Z}_{\ge 0}}$ *with* $\overline{\mathbf{x}}^{(0)} \ge \mathbf{x}^*$.

Remark 2. *Both* PI *and* VI *algorithms converge for SSGs. We remark that the convergence of* VI *is established for iterates from below and from above. Specifically, after every* n *steps,* VI *achieves linear convergence with a contraction factor* $\tilde{\gamma}^n \in [0, 1)$.

As a final remark, although PI *and* VI *algorithm perform well in practice [14], they both run in exponential time in the worse case.*[6] *To get any polynomial bound for SSGs requires a major breakthrough in the field.*

3 Equivalence Between LEMM Subclasses

In this section, we will show reductions between several LEMM subclasses. Before preceeding with the reductions, we first state a few equivalent characterization of Condition C1+, the proof of which can be found, for instance, in [4, Lemma 9].

Proposition 4. *Given an LEMM with* ($S_{\min}$, $S_{\max}$, S_{aff}, n, $\mathcal{N}$, $\mathbf{q}$, $\mathbf{b}$)*, the following conditions are equivalent:*

- *Condition C1+ holds.*
- $\max_{\mathbf{Q}\in\mathcal{Q}} \varrho(|\mathbf{Q}|) < 1$.
- *There exists* $\mathbf{v} \in \mathbb{R}^n_{\ge 1}$ *such that* $\mathbf{v} \ge |\mathbf{Q}|\mathbf{v} + \mathbf{1}$ *for all* $\mathbf{Q} \in \mathcal{Q}$.
- *For all* $\mathbf{c} \in \mathbb{R}^n$*, there exists* $\mathbf{v}^{(\mathbf{c})} \in \mathbb{R}^n_{\ge 0}$ *such that* $\mathbf{v}^{(\mathbf{c})} \ge |\mathbf{Q}|\mathbf{v}^{(\mathbf{c})} + \mathbf{c}$ *for all* $\mathbf{Q} \in \mathcal{Q}$.

The following lemma uses Proposition 4 to show that under Condition C1+ (or under Conditions C1 and C2), the Condition C3 can be assumed without loss of generality. The main idea of the reduction is to use the vector $\mathbf{v}^{(\mathbf{b})}$ from Proposition 4 to contract the row sum.

Lemma 1. *Any LEMM under* {C1+} *can be reduced to an LEMM under* {C1+,C3} *of the same size; any LEMM under* {C1,C2} *can be reduced to an LEMM under* {C1,C2,C3} *of the same size.*

Proof. Consider an LEMM given by ($S_{\min}, S_{\max}, S_{\text{aff}}, n, \mathcal{N}, \mathbf{q}, \mathbf{b}$) that satisfies Conditions C1 and C2. By Proposition 4, there exists $\mathbf{v} = [v_1, \cdots, v_n] \in \mathbb{R}^n_{\ge 0}$ such that

$$\mathbf{v} \ge \mathbf{Qv} + \mathbf{b} \text{ for all } \mathbf{Q} \in \mathcal{Q}, \quad (7)$$

[6] In the analysis of VI, Eq. (6) is exponential for $\tilde{\gamma}$ close to 1.

and moreover, such a $\mathbf{v}$ can be found in polynomial time by solving the linear program. Let $\Lambda = [\text{diag}(\mathbf{v})]^{-1}$. Then, $\mathbf{x} \in \mathbb{R}^n$ is the solution of the original LEMM if and only if $\mathbf{y} = \Lambda\mathbf{x}$ is the solution of the new LEMM given by $(S_{\min}, S_{\max}, S_{\text{aff}}, n, \mathcal{N}, \mathbf{q}', \Lambda\mathbf{b})$, where $\mathbf{q}'_k = v_k^{-1}\Lambda^{-1}\mathbf{q}_k$, $k \in S_{\text{aff}}$.

Moreover, for all $k \in S_{\text{aff}}$, we have

$$\begin{aligned}(\mathbf{q}'_k)^\top \mathbf{1} + (\Lambda\mathbf{b})_k &= v_k^{-1}(\mathbf{q}_k^\top \Lambda^{-1}\mathbf{1} + b_k) \\ &\leq v_k^{-1}(\mathbf{q}_k^\top \mathbf{v} + b_k) \leq v_k^{-1} v_k = 1\,,\end{aligned} \tag{8}$$

where the last inequality follows from Eq. (7). Hence, the new LEMM satisfies Condition C3. Hence, we have reduced the original LEMM under {C1,C2} to an LEMM (of the same size) satisfying {C1,C2,C3}.

Following similar steps, any LEMM under {C1+} can also be reduced to an LEMM (of the same size) satisfying {C1+,C3}. □

Now, we are able to obtain the equivalence results between subclasses of LEMMs in Theorem 1. The detailed proof can be found in Section B. The main proof steps of Item 1 of the theorem are to assume "sum upto one" by Lemma 1, make both positive and negative copies for each variable, and rewrite the max operators as the negative of min operator. The Item 2 of the theorem follows, again, from Lemma 1 and the equivalence result from the literature [7].

Theorem 1 (Equivalence between LEMM subclasses). *We obtain the following equivalence:*

1. *There is a linear reduction between the LEMM decision problems under* {C1+}, {C1+,C3}, {C1+,C4}, *and* {C1+,C3,C4}.
2. *There is a linear reduction between the LEMM decision problems under* {C1,C2}, {C2,C3}, *and* {C1,C2,C3}.

Remark 3. *From Item 1 in Theorem 1, we know that, for solving LEMMs under Condition C1+, assuming Conditions C3 and C4 will not make the problem easier. Therefore, we put the four linearly equivalent LEMM subproblems* {C1+}, {C1+,C3}, {C1+,C4}, *and* {C1+,C3,C4} *in the same row in Fig. 1.*

From Item 2 in Theorem 1, we know that the LEMM under {C1,C2} *is equivalent to the LEMM under* {C2,C3}, *which implies the linear equivalence between stochastic games and halting branching process. Therefore, we put the three linearly equivalent LEMM subproblems* {C1,C2}, {C2,C3}, *and* {C1,C2,C3} *in the same row in Fig. 1.*

It should also be noted that our reductions are of linear size, which implies any subexponential-time complexity result for SSGs (for instance, [12]) holds for the class of LEMMs under {C1,C2} *as well.*

The equivalence results established in Theorem 1 have simplified the complexity hierarchy of the LEMM decision problems in the original study of [4]. This simplification leads to a clearer structure containing only two new subclasses: the absolutely halting LEMMs under {C1+} and the halting LEMMs under {C1}. These two key subclasses are known to generalize SSGs yet still belong to the complexity class UP ∩ coUP.

4 Algorithms for LEMM Subclasses

In Section 4.1 and Section 4.2, we study the algorithms for the absolutely halting LEMMs and the halting LEMMs, respectively.

4.1 Algorithms for Absolutely Halting LEMMs

In this section, we study algorithms for LEMMs under Condition C1+. We first construct a counterexample showing that the PI algorithm may diverge for this class due to the negative coefficients. Then, we analyse the convergence of VI algorithm via a new preconditioning technique, connecting the convergence rate to spectral gap. Finally, we apply our new VI analysis approach on the fundamental subproblems of SSGs (and MDPs). It leads to surprising improvements of the long-standing runtime analyses of VI and PI, which is a key result of this work.

Example 1 (Divergence of PI). *Consider the LEMM given by:*

$$\begin{cases} x_1 = \max\{x_4, x_5\}, \\ x_2 = \max\{x_6, x_7\}, \\ x_3 = \max\{x_8, x_9\}, \\ x_4 = -0.2x_2 + 0.2x_3 + 0.25, \\ x_5 = 0.3x_1 - 0.6x_3 + 0.25, \\ x_6 = -0.5x_1 + 0.25, \\ x_7 = 0.3x_1 + 0.1x_2 - 0.3x_3 + 0.25, \\ x_8 = -0.2x_1 + 0.4x_2 + 0.3, \\ x_9 = -0.7x_2 + 0.3x_3 + 0.3. \end{cases}$$

The above LEMM satisfies Conditions C1+, C3 and C4. Consider the algorithm $\mathsf{PImax}(\mathbf{Q}_{\max}^{(0)})$ *with*

$$\mathbf{Q}_{\max}^{(0)} = [\mathbf{e}_4, \mathbf{e}_6, \mathbf{e}_8]^\top,$$

then

$$\begin{aligned} \mathbf{Q}_{\max}^{(1)} &= \pi_{\max}(\mathbf{x}_{\max}(\mathbf{Q}_{\max}^{(0)})) = [\mathbf{e}_4, \mathbf{e}_7, \mathbf{e}_8]^\top, \\ \mathbf{Q}_{\max}^{(2)} &= \pi_{\max}(\mathbf{x}_{\max}(\mathbf{Q}_{\max}^{(1)})) = [\mathbf{e}_4, \mathbf{e}_7, \mathbf{e}_9]^\top, \\ \mathbf{Q}_{\max}^{(3)} &= \pi_{\max}(\mathbf{x}_{\max}(\mathbf{Q}_{\max}^{(2)})) = [\mathbf{e}_5, \mathbf{e}_7, \mathbf{e}_9]^\top, \\ \mathbf{Q}_{\max}^{(4)} &= \pi_{\max}(\mathbf{x}_{\max}(\mathbf{Q}_{\max}^{(3)})) = [\mathbf{e}_5, \mathbf{e}_7, \mathbf{e}_8]^\top, \\ \mathbf{Q}_{\max}^{(5)} &= \pi_{\max}(\mathbf{x}_{\max}(\mathbf{Q}_{\max}^{(4)})) = [\mathbf{e}_4, \mathbf{e}_7, \mathbf{e}_8]^\top, \\ \mathbf{Q}_{\max}^{(6)} &= \pi_{\max}(\mathbf{x}_{\max}(\mathbf{Q}_{\max}^{(5)})) = [\mathbf{e}_4, \mathbf{e}_6, \mathbf{e}_8]^\top, \\ &\cdots\cdots \end{aligned}$$

The above algorithm enters a loop.

Next, we show that the VI algorithm still converges under Condition C1+. For the SSG subclass, the classic analysis of VI assumes starting from *a lower bound* (or *an upper bound*) of the solution, and then shows that the sequence monotonically increases (or decreases) and converges to the solution, as stated in Item 2 in Proposition 3. For LEMMs under Condition C1+, however, without non-negativity (Condition C2), the VI iterates are not necessarily monotonic. Even worse, the classic halting conditioning $\tilde{\gamma}$ for SSGs (cf. Eq. (5)) is not well-defined for the generalized LEMMs, as shown in the following example:

Example 2 (Ill-defined classic conditioning). *Consider for instance* $\mathcal{Q} = \{\mathbf{Q}\}$ *with the matrix*

$$\mathbf{Q} = \begin{bmatrix} 0.5 & 0.7 \\ 0 & 0.7 \end{bmatrix}$$

whose spectral radius $\varrho(\mathbf{Q}) = 0.7 < 1$, *yet we have* $\tilde{\gamma}^2 = \|\mathbf{Q}^2\mathbf{1}\|_\infty = 1.49 > 1$.

Hence, we need a different convergence analysis for VI algorithm under Condition C1+. To this end, we propose a new analysis of VI with a novel preconditioning technique.

Preconditioning. We define a new halting conditioning

$$\gamma \triangleq \max_{\mathbf{Q}\in\mathcal{Q}} \varrho(|\mathbf{Q}|) < 1, \tag{9}$$

and refer to $(1-\gamma)$ as *spectral gap*.

For any real number $\alpha \in (0,1)$, let

$$\gamma^{(\alpha)} \triangleq \gamma + \alpha(1-\gamma) \in (\gamma, 1). \tag{10}$$

Moreover, let $\mathbf{v}^{(\alpha)} \in \mathbb{R}^n_{\geq 0}$ such that

$$\mathbf{v}^{(\alpha)} \triangleq \inf \left\{\mathbf{v} \in \mathbb{R}^n_{\geq 0} \middle| \, \gamma^{(\alpha)}\mathbf{v} \geq |\mathbf{Q}|\mathbf{v} + \mathbf{1}, \text{ for all } \mathbf{Q} \in \mathcal{Q}\right\}, \tag{11}$$

which is well defined in view of Proposition 4. Then, let

$$\Lambda^{(\alpha)} \triangleq [\mathrm{diag}(\mathbf{v}^{(\alpha)})]^{-1}. \tag{12}$$

The following lemma shows some useful properties of the new conditionings we introduced, as well as their relations to the classic SSG conditioning $\tilde{\gamma}$ (cf. Eq. (5)).

Lemma 2. *Consider an LEMM with $(S_{\min}, S_{\max}, S_{\text{aff}}, n, \mathcal{N}, \mathbf{q}, \mathbf{b})$ under Condition C1+. Let $\alpha \in (0,1)$. The following properties hold:*

(1) $1-\gamma^{(\alpha)} = (1-\alpha)(1-\gamma)$;
(2) $\|\Lambda^{(\alpha)}\mathbf{Q}(\Lambda^{(\alpha)})^{-1}\|_\infty \leq \gamma^{(\alpha)}$, *for all* $\mathbf{Q} \in \mathcal{Q}$;
(3) $\left\|\Lambda^{(\alpha)}(\mathbf{I}-\mathbf{Q})^{-1}(\Lambda^{(\alpha)})^{-1}\right\|_\infty \leq \frac{1}{1-\gamma^{(\alpha)}}$, *for all* $\mathbf{Q} \in \mathcal{Q}$;
(4) Under Condition C2, we have $\gamma \leq \tilde{\gamma}$;
(5) Suppose all input numbers are rationals. Under Conditions C2 and C3, we have $\|\mathbf{v}^{(\alpha)}\|_\infty \leq \alpha^{-n}D^n n(\gamma^{(\alpha)}+1)^{n-1}$.

Proof. (1) $1-\gamma^{(\alpha)} = 1-\gamma-\alpha(1-\gamma) = (1-\alpha)(1-\gamma)$.

(2) For all $\mathbf{Q} \in \mathcal{Q}$, we have for all $i \in [n]$,

$$\begin{aligned}\sum_{j=1}^{n}(\Lambda^{(\alpha)}|\mathbf{Q}|(\Lambda^{(\alpha)})^{-1})_{i,j} &= \frac{1}{v_i^{(\alpha)}}\Big(\sum_{j=1}^{n}|\mathbf{Q}|_{i,j}v_j^{(\alpha)}\Big) \\ &\leq \frac{1}{v_i^{(\alpha)}}(\gamma^{(\alpha)}v_i^{(\alpha)}) = \gamma^{(\alpha)},\end{aligned}$$

and therefore, $\|\Lambda^{(\alpha)}\mathbf{Q}(\Lambda^{(\alpha)})^{-1}\|_\infty \leq \gamma^{(\alpha)}$.

(3) For all $\mathbf{Q} \in \mathcal{Q}$, we have

$$\begin{aligned}\left\|\Lambda^{(\alpha)}(\mathbf{I}-\mathbf{Q})^{-1}(\Lambda^{(\alpha)})^{-1}\right\|_\infty &\leq \sum_{k=0}^{\infty}\left\|\Lambda^{(\alpha)}\mathbf{Q}^k(\Lambda^{(\alpha)})^{-1}\right\|_\infty \\ = \sum_{k=0}^{\infty}\left\|\left(\Lambda^{(\alpha)}\mathbf{Q}(\Lambda^{(\alpha)})^{-1}\right)^k\right\|_\infty &\leq \sum_{k=0}^{\infty}\left\|\Lambda^{(\alpha)}\mathbf{Q}(\Lambda^{(\alpha)})^{-1}\right\|_\infty^k \\ \leq \sum_{k=0}^{\infty}(\gamma^{(\alpha)})^k &= \frac{1}{1-\gamma^{(\alpha)}}.\end{aligned}$$

(4) Under Condition C2, by the definitions of γ and $\tilde{\gamma}$, we have the following inequalities

$$\gamma = \max_{\mathbf{Q}\in\mathcal{Q}} \varrho(\mathbf{Q}) \leq \max_{\mathbf{Q}\in\mathcal{Q}} \|\mathbf{Q}^n\|_\infty^{\frac{1}{n}} \leq \tilde{\gamma}.$$

(5) Under Condition C2, by the definition of $\mathbf{v}^{(\alpha)}$, there exists $\mathbf{Q}^{(\alpha)} \in \mathcal{Q}$ such that

$$\gamma^{(\alpha)}\mathbf{v}^{(\alpha)} = \mathbf{Q}^{(\alpha)}\mathbf{v}^{(\alpha)} + \mathbf{1}.$$

Then, we have

$$\begin{aligned}\|\mathbf{v}^{(\alpha)}\|_\infty &= \left\|(\gamma^{(\alpha)}\mathbf{I}-\mathbf{Q}^{(\alpha)})^{-1}\right\|_\infty \\ &= \left|\det(\gamma^{(\alpha)}\mathbf{I}-\mathbf{Q}^{(\alpha)})\right|^{-1}\|\mathbf{M}^{(\alpha)}\|_\infty \\ &\leq \left|\det(\gamma^{(\alpha)}\mathbf{I}-\mathbf{Q}^{(\alpha)})\right|^{-1}\cdot n\cdot \max_{1\leq i,j\leq n}\left|M_{ij}^{(\alpha)}\right|.\end{aligned} \tag{13}$$

where $\mathbf{M}^{(\alpha)} = \left(M_{ij}^{(\alpha)}\right) \in \mathbb{R}^{n\times n}$ denote the adjugate matrix of $\left(\gamma^{(\alpha)}\mathbf{I}-\mathbf{Q}^{(\alpha)}\right)$.

Under Condition C3, we have for all $1 \leq i, j \leq n$,

$$\left|M_{ij}^{(\alpha)}\right| = \left|\det\left((\gamma^{(\alpha)}\mathbf{I}-\mathbf{Q}^{(\alpha)})_{-ji}\right)\right| \leq (\gamma^{(\alpha)}+1)^{n-1}. \tag{14}$$

Moreover, we have

$$\left|\det(\gamma^{(\alpha)}\mathbf{I}-\mathbf{Q}^{(\alpha)})\right| \geq \alpha^n\left|\det(\mathbf{I}-\mathbf{Q}^{(\alpha)})\right| \geq \alpha^n D^{-n}. \tag{15}$$

Finally, the desired inequality follows from plugging Eqs. (14) and (15) into Eq. (13). □

We are now ready to show a key lemma for our convergence analysis of VI. In particular, with the new conditioning in Eqs. (9) to (12), we can show the contraction of value iteration operator in *one step* under *the* $\Lambda^{(\alpha)}$*-preconditioned norm*, with which we can establish the convergence result of VI starting from *an arbitrary point*.

Lemma 3 (One-step contraction of $v(\cdot)$). *Consider an LEMM with $(S_{\min}, S_{\max}, S_{\text{aff}}, n, \mathcal{N}, \mathbf{q}, \mathbf{b})$ under Condition C1+. Let $\alpha \in (0,1)$. For all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$,*

$$\|\Lambda^{(\alpha)}(v(\mathbf{x})-v(\mathbf{y}))\|_\infty \leq \gamma^{(\alpha)}\|\Lambda^{(\alpha)}(\mathbf{x}-\mathbf{y})\|_\infty.$$

Proof. For all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, let $v(\mathbf{x}) = \mathbf{Q}^{(\mathbf{x})}\mathbf{x}+\mathbf{b}$ and $v(\mathbf{y}) = \mathbf{Q}^{(\mathbf{y})}\mathbf{y}+\mathbf{b}$, where $\mathbf{Q}^{(\mathbf{x})}, \mathbf{Q}^{(\mathbf{y})} \in \mathcal{Q}$. Then, for all $\mathbf{Q} \in \mathcal{Q}$, we have

$$\mathbf{Q}^{(\mathbf{x})}_{i,\cdot}\mathbf{x} \leq \mathbf{Q}_{i,\cdot}\mathbf{x}, \quad \mathbf{Q}^{(\mathbf{y})}_{i,\cdot}\mathbf{y} \leq \mathbf{Q}_{i,\cdot}\mathbf{y}, \quad \text{for all } i \in S_{\min}, \text{ and}$$
$$\mathbf{Q}^{(\mathbf{x})}_{j,\cdot}\mathbf{x} \geq \mathbf{Q}_{j,\cdot}\mathbf{x}, \quad \mathbf{Q}^{(\mathbf{y})}_{j,\cdot}\mathbf{y} \geq \mathbf{Q}_{j,\cdot}\mathbf{y}, \quad \text{for all } j \in S_{\max}.$$

Construct the matrix $\mathbf{Q}^{(\mathbf{x},\mathbf{y})} \in \mathcal{Q}$ as follows: (i) take the rows corresponding to min variables from $\mathbf{Q}^{(\mathbf{y})}$; (ii) take the rows corresponding to max variables from $\mathbf{Q}^{(\mathbf{x})}$; and (iii) take the remaining rows as $\mathbf{Q}_{\text{aff}}$. That is: $\mathbf{Q}^{(\mathbf{x},\mathbf{y})}_{i,\cdot} = \mathbf{Q}^{(\mathbf{y})}_{i,\cdot}$ for $i \in S_{\min}$; $\mathbf{Q}^{(\mathbf{x},\mathbf{y})}_{j,\cdot} = \mathbf{Q}^{(\mathbf{x})}_{j,\cdot}$ for $j \in S_{\max}$; and $\mathbf{Q}^{(\mathbf{x},\mathbf{y})}_{k,\cdot} = \mathbf{Q}^{(\mathbf{x})}_{k,\cdot} = \mathbf{Q}^{(\mathbf{y})}_{k,\cdot}$ for $k \in S_{\text{aff}}$.

Then, we have

$$\begin{aligned}\Lambda^{(\alpha)}(v(\mathbf{x})-v(\mathbf{y})) &= \Lambda^{(\alpha)}\left((\mathbf{Q}^{(\mathbf{x})}\mathbf{x}+\mathbf{b})-(\mathbf{Q}^{(\mathbf{y})}\mathbf{y}+\mathbf{b})\right) \\ &= \Lambda^{(\alpha)}\left(\mathbf{Q}^{(\mathbf{x})}\mathbf{x}-\mathbf{Q}^{(\mathbf{y})}\mathbf{y}\right) \overset{(*)}{\geq} \Lambda^{(\alpha)}\mathbf{Q}^{(\mathbf{x},\mathbf{y})}(\mathbf{x}-\mathbf{y}) \\ &\geq -\|\Lambda^{(\alpha)}\mathbf{Q}^{(\mathbf{x},\mathbf{y})}(\mathbf{x}-\mathbf{y})\|_\infty \\ &= -\|\Lambda^{(\alpha)}\mathbf{Q}^{(\mathbf{x},\mathbf{y})}\left(\Lambda^{(\alpha)}\right)^{-1}\Lambda^{(\alpha)}(\mathbf{x}-\mathbf{y})\|_\infty \\ &\geq -\|\Lambda^{(\alpha)}\mathbf{Q}^{(\mathbf{x},\mathbf{y})}\left(\Lambda^{(\alpha)}\right)^{-1}\|_\infty\|\Lambda^{(\alpha)}(\mathbf{x}-\mathbf{y})\|_\infty \\ &\overset{(**)}{\geq} -\gamma^{(\alpha)}\|\Lambda^{(\alpha)}(\mathbf{x}-\mathbf{y})\|_\infty,\end{aligned} \tag{16}$$

where (*) follows from the construction of $\mathbf{Q}^{(\mathbf{x},\mathbf{y})}$ and (**) follows from Item 2 in Lemma 2.

Similarly, we can show that for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$,

$$\Lambda^{(\alpha)}(v(\mathbf{x})-v(\mathbf{y})) \leq \gamma^{(\alpha)}\|\Lambda^{(\alpha)}(\mathbf{x}-\mathbf{y})\|_\infty.$$

Therefore, we have

$$\|\Lambda^{(\alpha)}(v(\mathbf{x})-v(\mathbf{y}))\|_\infty \le \gamma^{(\alpha)}\|\Lambda^{(\alpha)}(\mathbf{x}-\mathbf{y})\|_\infty ,$$

for all $\mathbf{x},\mathbf{y}\in\mathbb{R}^n$. □

Finally, we conclude with the results for absolutely halting LEMMs in the following theorem.

Theorem 2 (Analyses for LEMM under {C1+}). *Consider an LEMM with $(S_{\min}, S_{\max}, S_{\mathrm{aff}}, n, \mathcal{N}, \mathbf{q}, \mathbf{b})$ under Condition C1+. Let $\mathbf{x}^*$ be its unique solution.*

(1) PImax or PImin may not converge.

(2) Let $\mathbf{x}^{(0)}\in\mathbb{R}^n$ be any given point. The following assertions hold for the iterates $(\mathbf{x}^{(t)})_{t\in\mathbb{Z}_{\ge 0}}$ in $\mathsf{VI}(\mathbf{x}^{(0)})$:

(a) For all $t\in\mathbb{Z}_{\ge0}$ and for all $\alpha\in(0,1)$, we have

$$\begin{aligned}\|\Lambda^{(\alpha)}(\mathbf{x}^{(t)}-\mathbf{x}^*)\|_\infty &\le \gamma^{(\alpha)}\|\Lambda^{(\alpha)}(\mathbf{x}^{(t-1)}-\mathbf{x}^*)\|_\infty\\ &\le \cdots \le (\gamma^{(\alpha)})^t\|\Lambda^{(\alpha)}(\mathbf{x}^{(0)}-\mathbf{x}^*)\|_\infty .\end{aligned}$$

(b) Suppose all input numbers are rationals. There exists

$$\begin{aligned}t < \inf_{\alpha\in(0,1)} \frac{1}{1-\gamma^{(\alpha)}}\big[n\log(2D) + \log(2\|\Lambda^{(\alpha)}(\mathbf{x}^{(0)}-\mathbf{x}^*)\|_\infty)\\ + \log\|\mathbf{v}^{(\alpha)}\|_\infty\big] + 1 ,\end{aligned}$$

such that $\mathbf{x}^{(t)}$ is an ε^*-accurate solution.

Proof. (1) follows from the counterexample in Example 1.
(2) follows from Lemma 3. □

Remark 4. *On the one hand, Item 1 in Theorem 2 shows that the classic PI algorithm diverges for absolutely halting LEMMs. Compared to Item 1 in Proposition 3, it also indicates that the absolutely halting LEMM seems to be a different problem class from SSGs.*

On the other hand, Item 2 shows the connection between the spectral gap $(1-\gamma)$ *and the convergence rate of* VI *for absolutely halting LEMMs, significantly generalizing the convergence result from SSGs. We also remark that it is well known in the context of Markov chains that the spectral gap determines mixing time, but to our knowledge, this relation has not been extended to the settings of MDPs or SSGs.*

In Theorem 2, we obtain a new analysis of value iteration that generalizes the convergence proof for SSGs (under {C1,C2,C3}) to absolutely halting LEMMs (under {C1+}). Next, we answer the following natural question: if limited to the well-studied setting of SSGs (or even MDPs), how is our new analysis compared to the classic result of Proposition 3?

New analysis for stochastic games

In this section, we discuss our new algorithmic results in the context of stochastic games (and eveh MDPs). First, we state our new analysis of VI applied back to SSGs, and demonstrate how it refines the existing analysis.

Corollary 3 (New VI analysis for SSGs). *Consider an LEMM with $(S_{\min}, S_{\max}, S_{\mathrm{aff}}, n, \mathcal{N}, \mathbf{q}, \mathbf{b})$ under conditions {C1,C2,C3}. Let $\mathbf{x}^{(0)}\in[0,1]^n$ be any given point and let $(\mathbf{x}^{(t)})_{t\in\mathbb{Z}_{\ge0}}$ be the iterates in $\mathsf{VI}(\mathbf{x}^{(0)})$. Suppose all input numbers are rationals. There exists*

$$t = O\Big(\frac{n}{1-\gamma}\log(2D)\Big)$$

such that $\mathbf{x}^{(t)}$ is an ε^*-accurate solution. Thus, VI *is a polynomial time algorithm when γ is bounded away from 1.*

Proof. This is implied by Item 2 in Theorem 2 and Item 5 in Lemma 2. □

Example 3 (γ improves $\tilde{\gamma}$). *For every $n\in\mathbb{Z}_{>1}$, let*

$$\mathbf{Q}(n) = \begin{pmatrix} \delta & 1-\delta & 0 & \dots & \dots & 0\\ 0 & \delta & 1-\delta & 0 & \dots & 0\\ \vdots & 0 & \ddots & \ddots & \ddots & \vdots\\ \vdots & & \ddots & \ddots & \ddots & 0\\ 0 & \dots & \dots & 0 & \delta & 1-\delta\\ 0 & \dots & \dots & \dots & 0 & \delta \end{pmatrix} \in \mathbb{R}^{n\times n} .$$

We have

$$1-\varrho(\mathbf{Q}(n)) = 1-\delta .$$

Moreover, we have

$$\begin{aligned}\|(\mathbf{Q}(n))^n\|_\infty &= \Big\|\sum_{k=0}^{n-1}\binom{n}{k}\delta^{n-k}(1-\delta)^k(\mathrm{S}(n))^k\Big\|_\infty\\ &= \sum_{k=0}^{n-1}\binom{n}{k}\delta^{n-k}(1-\delta)^k\\ &= 1-(1-\delta)^n ,\end{aligned}$$

and then,

$$\begin{aligned}1-\|(\mathbf{Q}(n))^n\|_\infty^{\frac{1}{n}} &= 1-(1-(1-\delta)^n)^{\frac{1}{n}}\\ &= 1-\exp\Big(\frac{1}{n}\ln(1-(1-\delta)^n)\Big)\\ &\le -\frac{1}{n}\ln(1-(1-\delta)^n)\\ &\le \frac{1}{n}\cdot\frac{(1-\delta)^n}{1-(1-\delta)^n} ,\end{aligned}$$

where the last inequality follows from $-\ln(1-x)\le\frac{x}{1-x}$ for $x\in(0,1)$. Now, fixing $\delta=0.5$, we have

$$\begin{aligned}1-\|(\mathbf{Q}(n))^n\|_\infty^{\frac{1}{n}} &\le \frac{1}{n}\cdot\frac{0.5^n}{1-0.5^n}\\ &\le \frac{1}{n\cdot(2^n-1)} ,\end{aligned}$$

and then,

$$\frac{1-\varrho(\mathbf{Q}(n))}{1-\|(\mathbf{Q}(n))^n\|_\infty^{\frac{1}{n}}} \ge \frac{1}{2}n\cdot(2^n-1) .$$

Therefore, for $\mathcal{Q}(n)=\{\mathbf{Q}(n)\}$, we have $(1-\gamma)$ is exponentially larger than $(1-\tilde{\gamma})$.

Remark 5. *In view of Item 4 in Lemma 2, the convergence rate in Corollary 3 is consistently no worse than the classic $O\big(\frac{n}{1-\tilde{\gamma}}\log(2D)\big)$ rate in Eq. (6) up to constants. Moreover, our new rate can be exponentially faster, as demonstrated in Example 3. Our new analysis of* VI*, based on the spectral gap, generalizes to absolutely halting LEMMs; and perhaps surprisingly, it simultaneously yields improvements over existing state-of-the-art analyses for well-studied subproblems. Let us conclude with the new runtime result of* VI*: for both halting MDPs and stochastic games* with reachability objectives, VI *is a polynomial time algorithm (provided γ is bounded away from 1).*

Now, we show a new analysis of PI for SSGs. This is obtained by combining our new convergence rate of VI and a well-known result from the literature—PI iterates move faster than VI iterates.

PROPOSITION 5 ([5, 6, 8]). *Consider an LEMM with* $(S_{\min}, S_{\max}, S_{\text{aff}}, n, \mathcal{N}, \mathbf{q}, \mathbf{b})$ *under conditions* {C1,C2,C3}. *Let* $\mathbf{x}^*$ *be its unique solution, let* $\mathbf{Q}_{\max}^{(0)} \in \mathcal{Q}_{\max}$ *be any given max policy, let* $(\underline{\mathbf{x}}^{(t)})_{t\in\mathbb{Z}_{\geq 0}}$ *be the iterates in* $\mathsf{PImax}(\mathbf{Q}_{\max}^{(0)})$*, and let* $(\mathbf{x}^{(t)})_{t\in\mathbb{Z}_{\geq 0}}$ *be the iterates in* $\mathsf{VI}(\underline{\mathbf{x}}^{(0)})$. *For all* $t \in \mathbb{Z}_{\geq 0}$*, the following assertions hold:*

(1) $\underline{\mathbf{x}}^{(t)} \leq \underline{\mathbf{x}}^{(t+1)}$,
(2) $\underline{\mathbf{x}}^{(t)} \leq \mathbf{x}^*$*, and*
(3) $\mathbf{x}^{(t)} \leq \underline{\mathbf{x}}^{(t)}$.

THEOREM 4 (NEW PI ANALYSIS FOR SSGS). *Consider an LEMM with* $(S_{\min}, S_{\max}, S_{\text{aff}}, n, \mathcal{N}, \mathbf{q}, \mathbf{b})$ *under conditions* {C1,C2,C3}. *Let* $\mathbf{Q}_{\max}^{(0)} \in \mathcal{Q}_{\max}$ *be any given max policy, and let* $(\underline{\mathbf{x}}^{(t)})_{t\in\mathbb{Z}_{\geq 0}}$ *be the iterates in* $\mathsf{PImax}(\mathbf{Q}_{\max}^{(0)})$. *There exists*

$$t = O\Big(\frac{m}{1-\gamma}\log\big(\frac{1}{1-\gamma}\big)\Big)$$

such that $\underline{\mathbf{x}}^{(t)}$ *is the solution. Similar assertion holds for* $\mathsf{PImin}(\mathbf{Q}_{\min}^{(0)})$ *for any given* $\mathbf{Q}_{\min}^{(0)} \in \mathcal{Q}_{\min}$. *Thus,* PI *is a strongly polynomial-time algorithm when* γ *is bounded away from* 1.

PROOF OF THEOREM 4. Let $\alpha \in (0,1)$ be a fixed constant. We define the following *reduced vector*:

$$\mathbf{r}^{\mathbf{Q}} = \Lambda^{(\alpha)}(\mathbf{x}^* - \mathbf{Q}\mathbf{x}^* - \mathbf{b}), \text{ for all } \mathbf{Q} \in \mathcal{Q}\,.$$

If $\mathbf{r}^{\mathbf{Q}^{(0)}} = \mathbf{0}$, then $\underline{\mathbf{x}}^{(0)}$ is the solution, and the theorem trivially holds. Now, let us assume $\mathbf{r}^{\mathbf{Q}^{(0)}} \neq \mathbf{0}$.

For all $k \in \mathbb{Z}_{\geq 0}$, since $\underline{\mathbf{x}}^{(k)} = \mathbf{x}_{\max}(\mathbf{Q}_{\max}^{(k)})$, let $\mathbf{Q}^{(k)} \in \mathcal{Q}$ such that

$$\underline{\mathbf{x}}^{(k)} = \mathbf{Q}^{(k)}\underline{\mathbf{x}}^{(k)} + \mathbf{b}\,. \tag{17}$$

For all $\alpha \in (0,1)$, we have

$$\mathbf{r}^{\mathbf{Q}^{(k)}} = \Lambda^{(\alpha)}(\mathbf{x}^* - \mathbf{Q}^{(k)}\mathbf{x}^* - \mathbf{b}) = \Lambda^{(\alpha)}(\mathbf{I} - \mathbf{Q}^{(k)})(\mathbf{x}^* - \underline{\mathbf{x}}^{(k)})$$

due to Eq. (17), and

$$\mathbf{x}^* - \underline{\mathbf{x}}^{(k)} \geq 0$$

due to Item 2 in Proposition 5. Therefore,

$$-\Lambda^{(\alpha)}\mathbf{Q}^{(k)}(\mathbf{x}^* - \underline{\mathbf{x}}^{(k)}) \leq \mathbf{r}^{\mathbf{Q}^{(k)}} \leq \Lambda^{(\alpha)}(\mathbf{x}^* - \underline{\mathbf{x}}^{(k)}),$$

and in view of

$$\begin{aligned}
&\|\Lambda^{(\alpha)}\mathbf{Q}^{(k)}(\mathbf{x}^* - \underline{\mathbf{x}}^{(k)})\|_\infty \\
&\quad\leq \|\Lambda^{(\alpha)}\mathbf{Q}^{(k)}(\Lambda^{(\alpha)})^{-1}\|_\infty \cdot \|\Lambda^{(\alpha)}(\mathbf{x}^* - \underline{\mathbf{x}}^{(k)})\|_\infty \\
&\quad\leq \gamma^{(\alpha)}\|\Lambda^{(\alpha)}(\mathbf{x}^* - \underline{\mathbf{x}}^{(k)})\|_\infty \\
&\quad\leq \|\Lambda^{(\alpha)}(\mathbf{x}^* - \underline{\mathbf{x}}^{(k)})\|_\infty,
\end{aligned}$$

we have

$$\|\mathbf{r}^{\mathbf{Q}^{(k)}}\|_\infty \leq \|\Lambda^{(\alpha)}(\mathbf{x}^* - \underline{\mathbf{x}}^{(k)})\|_\infty\,.$$

Let us consider an auxiliary sequence $(\mathbf{x}^{(t)})_{t=0}^k$ as the iterates in $\mathsf{VI}(\underline{\mathbf{x}}^{(0)})$. Then, we have

$$\begin{aligned}
\|\mathbf{r}^{\mathbf{Q}^{(k)}}\|_\infty &\leq \|\Lambda^{(\alpha)}(\mathbf{x}^* - \underline{\mathbf{x}}^{(k)})\|_\infty \overset{(*)}{\leq} \|\Lambda^{(\alpha)}(\mathbf{x}^* - \mathbf{x}^{(k)})\|_\infty \\
&\overset{(**)}{\leq} (\gamma^{(\alpha)})^k\|\Lambda^{(\alpha)}(\mathbf{x}^* - \mathbf{x}^{(0)})\|_\infty \\
&= (\gamma^{(\alpha)})^k\|\Lambda^{(\alpha)}(\mathbf{x}^* - \underline{\mathbf{x}}^{(0)})\|_\infty \\
&\leq (\gamma^{(\alpha)})^k\|\Lambda^{(\alpha)}(\mathbf{I} - \mathbf{Q}^{(0)})^{-1}(\Lambda^{(\alpha)})^{-1}\|_\infty\|\mathbf{r}^{\mathbf{Q}^{(0)}}\|_\infty \\
&\overset{(***)}{\leq} \frac{(\gamma^{(\alpha)})^k}{1-\gamma^{(\alpha)}}\|\mathbf{r}^{\mathbf{Q}^{(0)}}\|_\infty,
\end{aligned}$$

where (*) follows from Item 3 in Proposition 5, (**) follows from Item 2 in Theorem 2, and (***) follows from Item 3 in Lemma 2.

Let $s_0 \in \arg\max_{i\in[n]}|\mathbf{r}_i^{\mathbf{Q}^{(0)}}|$. For all

$$k \geq \Big\lceil \log_{1/\gamma^{(\alpha)}}\big(\frac{1}{1-\gamma^{(\alpha)}}\big)\Big\rceil \triangleq K,$$

we have

$$|\mathbf{r}_{s_0}^{\mathbf{Q}^{(k)}}| \leq \|\mathbf{r}^{\mathbf{Q}^{(k)}}\|_\infty \leq \frac{(\gamma^{(\alpha)})^k}{1-\gamma^{(\alpha)}}\|\mathbf{r}^{\mathbf{Q}^{(0)}}\|_\infty < \|\mathbf{r}^{\mathbf{Q}^{(0)}}\|_\infty = |\mathbf{r}_{s_0}^{\mathbf{Q}^{(0)}}|\,,$$

and thus,

$$\mathbf{Q}_{s_0,\cdot}^{(k)} \neq \mathbf{Q}_{s_0,\cdot}^{(0)}.$$

Therefore, if $\underline{\mathbf{x}}^{(0)}$ is not the solution, then $\mathbf{Q}^{(0)}$ contains a row $\mathbf{Q}_{s_0,\cdot}^{(0)}$ that will be eliminated from any $\mathbf{Q}_{s_0,\cdot}^{(k)}$ where $k \geq K$.

Finally, we can repeat the same argument for $\underline{\mathbf{x}}^{(0)}, \underline{\mathbf{x}}^{(K)}, \underline{\mathbf{x}}^{(2K)}, \cdots$ There are at most $m - n_1 - n_2$ row choices to eliminate, and therefore, there exists

$$\begin{aligned}
t &\leq (m - n_1 - n_2)\cdot K \\
&= O\Big(\frac{m}{1-\gamma}\log\big(\frac{1}{1-\gamma}\big)\Big)
\end{aligned}$$

such that $\underline{\mathbf{x}}^{(t)}$ is the solution. □

Finally, we state an immediate consequence of Theorem 4 for stochastic games with discounted-sum objectives (while more detail of the discounted-sum games are deferred to Section C):

COROLLARY 5 (NEW PI ANALYSIS FOR DISCOUNTED-SUM GAMES). *Consider a two-player turn-based zero-sum stochastic game with discounted-sum objective where the discount factor* $\gamma < 1$. *Let* m *be the number of actions. Then,* PI *is a strongly polynomial time algorithm that converges in no more than* $O\big(\frac{m}{1-\gamma}\log(\frac{1}{1-\gamma})\big)$ *iterations.*

REMARK 6. *We give a technical remark on Theorem 4 and Corollary 5. The proof generally follows the framework of [8, 16, 20]. However, our analysis, based on reduced vector, is more general and streamlined. On the one hand, for stochastic games with discounted-sum objectives, while the state-of-the-art strongly polynomial time bound in [8, Theorem 7.5] is given by*

$$t = O\Big(\frac{m}{1-\gamma}\log\big(\frac{n}{1-\gamma}\big)\Big),$$

our result in Corollary 5 improves it by removing the $\log n$ *factor. On the other hand, we extend the strongly polynomial time complexity result for the first time from discounted-sum objectives to* reachability objectives. *Let us conclude with the new complexity result of the problem classes: for both halting MDPs and stochastic* *games* with reachability objectives, PI *is a strongly polynomial time algorithm (provided γ is bounded away from* 1*).*

4.2 Algorithms for Halting LEMMs

In this section, we study the algorithms for LEMMs under Condition C1. We first construct a counterexample showing that the VI algorithm can also diverge for this class.

Example 4 (Divergence of VI). *Consider the LEMM given by:*

$$\begin{cases} x_1 = \max\{x_3, x_4\}, \\ x_2 = \max\{x_5, x_6\}, \\ x_3 = -0.9x_1 + 1.8x_2 - 1.5, \\ x_4 = 0.5x_1 + 1.5x_2 - 1.5, \\ x_5 = -0.5x_1 - 1.0, \\ x_6 = -0.25x_1 - 0.25x_2 - 1.0. \end{cases}$$

The above LEMM satisfies Conditions C1, C3 and C4. In particular, Condition C1 can be verified by noting the following facts: (i) the spectral radius of $\mathbf{Q}$ *is less than* 1*, for all* $\mathbf{Q} \in \mathcal{Q}$*; and (ii) the matrices* $(\mathbf{Q} + \mathbf{I})$ *and* $(\mathbf{Q} - \mathbf{I})$ *are both non-singular, for all* $\mathbf{Q} \in \operatorname{conv} \mathcal{Q}$*.*

Consider the algorithm $\mathsf{VI}(\mathbf{x}^{(0)})$ *with*

$$\mathbf{x}^{(0)} = [3/7, 5/7, -3/5, -3/14, -17/14, -9/7]^\top,$$

then

$$\mathbf{x}^{(1)} = v(\mathbf{x}^{(0)}) = [-3/14, -17/14, -3/5, -3/14, -17/14, -9/7]^\top,$$
$$\mathbf{x}^{(2)} = v(\mathbf{x}^{(1)}) = [-3/14, -17/14, -489/140, -24/7, -21/28, -9/14]^\top,$$
$$\mathbf{x}^{(3)} = v(\mathbf{x}^{(2)}) = [-24/7, -9/14, -489/140, -24/7, -21/28, -9/14]^\top,$$
$$\mathbf{x}^{(4)} = v(\mathbf{x}^{(3)}) = [-24/7, -9/14, 3/7, -117/28, 5/7, 1/56]^\top,$$
$$\mathbf{x}^{(5)} = v(\mathbf{x}^{(4)}) = [3/7, 5/7, 3/7, -117/28, 5/7, 1/56]^\top,$$
$$\mathbf{x}^{(6)} = v(\mathbf{x}^{(5)}) = [3/7, 5/7, -3/5, -3/14, -17/14, -9/7]^\top,$$
$$\cdots\cdots$$

The above algorithm enters a loop.

Given that both PI and VI can diverge, now we revisit a less studied algorithm for stochastic games, Simple Policy Iteration [13] as well as its randomized variant.

(Randomized) Simple Policy Iteration. We consider Simple Policy Iteration (SPI) as well as its randomized variant Randomized Simple Policy Iteration (RandSPI). The pseudocodes of SPI and RandSPI are given in Algorithm 3 and Algorithm 4, respectively. The SPI algorithm runs iteratively: starting from certain choices of neighbors, the algorithm finds the maximum index at which the corresponding min or max equation is violated, and then updates the neighbor choice at that index. The RandSPI algorithm runs recursively: when there are k min and max variables, the algorithm iterates over $\mathcal{N}(k)$ for the neighbor choice of the kth min or max variable, and then recursively solves the new LEMM with $(k-1)$ min and max variables. For RandSPI, we require the order $\mathcal{I}(k)$ within each recursive call to be chosen *independently* and uniformly at random.

Theorem 6 (Analyses for LEMM under {C1}). *Consider an LEMM with* $(S_{\min}, S_{\max}, S_{\text{aff}}, n, \mathcal{N}, \mathbf{q}, \mathbf{b})$ *under Condition C1.*

(1) Neither PI *nor* VI *converges.*

(2) Let $\mathcal{I}(i)$ *be any permutation of* $\mathcal{N}(i)$*, for all* $i \in [n_1+n_2]$*. Then,* $\mathsf{SPI}(\mathcal{I})$ *returns the solution* $\mathbf{x}^*$ *in no more than* $\Pi_{i \in S_{\min} \cup S_{\max}} |\mathcal{N}(i)|$ *iterations.*

Algorithm 3 Simple Policy Iteration $\mathsf{SPI}(\mathcal{I})$

Require: $\mathcal{I}(i)$ is a permutation of $\mathcal{N}(i)$, for all $i \in [n_1 + n_2]$.

Let $\mathbf{Q} \leftarrow [\mathbf{e}_{\mathcal{I}(1)(1)}, \cdots, \mathbf{e}_{\mathcal{I}(n_1+n_2)(1)}, \mathbf{q}_{n_1+n_2+1}, \cdots, \mathbf{q}_n]^\top$
Let $\boldsymbol{\ell} \leftarrow (1, \cdots, 1) \in [|\mathcal{N}(1)|] \times \cdots \times [|\mathcal{N}(n_1+n_2)|]$
loop
  $\mathbf{x} \leftarrow (\mathbf{I} - \mathbf{Q})^{-1}\mathbf{b}$
  $\Gamma \leftarrow \{i \in S_{\min} \mid \exists k \in \mathcal{N}(i) \text{ such that } x_k < x_i\} \cup \{j \in S_{\max} \mid \exists l \in \mathcal{N}(j) \text{ such that } x_l > x_j\}$
  if Γ is empty **then**
    return x
  else
    $k \leftarrow \max \Gamma$
    if $\ell_k \neq |\mathcal{N}(k)|$ **then**
      $\ell_k \leftarrow \ell_k + 1$
    else
      $\ell_k \leftarrow 1$
    end if
    $\mathbf{Q}_{k,\cdot} \leftarrow \mathbf{e}^\top_{\mathcal{I}(k)(\ell_k)}$
  end if
end loop

Algorithm 4 Randomized Simple Policy Iteration $\mathsf{RandSPI}(S_{\min}, S_{\max}, S_{\text{aff}}, n, \mathcal{N}, \mathbf{q}, \mathbf{b})$

if $S_{\min} \cup S_{\max}$ is non-empty **then**
  Let $k \leftarrow |S_{\min}| + |S_{\max}|$
  Get a (new) uniform random permutation $\mathcal{I}(k)$ of $\mathcal{N}(k)$
  for $i = 1, \cdots, |\mathcal{N}(k)|$ **do**
    if $k \in S_{\max}$ **then**
      $S'_{\max} \leftarrow S_{\max} \setminus \{k\}$
    else
      $S'_{\min} \leftarrow S_{\min} \setminus \{k\}$
    end if
    $S'_{\text{aff}} \leftarrow S_{\text{aff}} \cup \{k\}$
    Let $\mathbf{q}'$ such that
$$\mathbf{q}'_j = \begin{cases} \mathbf{e}_{\mathcal{I}(k)(i)}, & j = k, \\ \mathbf{q}_j, & k+1 \le j \le n. \end{cases}$$
    $\mathbf{x} \leftarrow \mathsf{RandSPI}(S'_{\min}, S'_{\max}, S'_{\text{aff}}, n, \mathcal{N}, \mathbf{q}', \mathbf{b})$
    if $k \in S_{\max}$ and $x_k = \max_{j \in \mathcal{N}(k)} x_j$ **then**
      Break the loop and **return x**
    else if $k \in S_{\min}$ and $x_k = \min_{j \in \mathcal{N}(k)} x_j$ **then**
      Break the loop and **return x**
    end if
  end for
else
  Let $\mathbf{Q} \leftarrow [\mathbf{q}_1, \cdots, \mathbf{q}_n]^\top$
  return $(\mathbf{I} - \mathbf{Q})^{-1}\mathbf{b}$
end if

(3) RandSPI$(S_{\min}, S_{\max}, S_{\text{aff}}, n, \mathcal{N}, \mathbf{q}, \mathbf{b})$ *returns the solution* $\mathbf{x}^*$ *in no more than*

$$\frac{\Pi_{i\in S_{\min}\cup S_{\max}}(|\mathcal{N}(i)|+1)}{2^{n_1+n_2}}$$

recursive calls in expectation.

Proof. (1) The divergence of PI and VI follows from Examples 1 and 4 respectively.

(2) We prove that the algorithm returns a feasible solution in no more than $\Pi_{i\in S_{\min}\cup S_{\max}}|\mathcal{N}(i)|$ iterations by induction on n_1+n_2, the number of min and max variables.

(Base) For $n_1+n_2=0$, due to Condition C1, $(\mathbf{I}-\mathbf{Q})$ is invertible, so the algorithm returns the feasible solution of the linear system in one iteration.

(Induction) Assume the algorithm returns a feasible solution in no more than $\Pi_{i\in S_{\min}\cup S_{\max}}|\mathcal{N}(i)|$ iterations if there are (n_1+n_2-1) min and max variables in an LEMM. Then, consider the LEMM with (n_1+n_2) min and max variables satisfying Condition C1, where there is a unique solution $\mathbf{x}^*$. Let l be the smallest number in $[|\mathcal{N}(n_1+n_2)|]$ such that $\mathbf{x}^*_{n_1+n_2}=x^*_{\mathcal{I}(n_1+n_2)(l)}$. We notice that the loop will not break before ℓ_k being set to l: if a vector $\mathbf{x}$ satisfies the first (n_1+n_2-1) equations, the (n_1+n_2)-th equation will be violated (otherwise, $\mathbf{x}^*$ would not be the unique solution of the original LEMM). Then, consider ℓ_k being set to l. By induction, a vector $\mathbf{x}$ satisfying the first (n_1+n_2-1) equations will be obtained, and this $\mathbf{x}=\mathbf{x}^*$ will be returned as the feasible solution of the original LEMM (otherwise, one can construct a new LEMM by replacing the (n_1+n_2)th equation by $x_{n_1+n_2}=x_l$, which has two different feasible solutions $\mathbf{x}$ and $\mathbf{x}^*$). Moreover, the number of iterations is upper bounded by

$$l\cdot\Pi_{i\in[n_1+n_2-1]}|\mathcal{N}(i)|\le\Pi_{i\in[n_1+n_2]}|\mathcal{N}(i)|$$

from the above analysis.

(3) Since there exists a solution in the LEMM, for each k there exists an index $i\in|\mathcal{N}(k)|$ from which the algorithm will return the feasible solution. Let $\mathcal{T}(k)$ denote the number of recursive calls for an LEMM with k min and max variables. Since the order $\mathcal{I}(k)$ in Algorithm 4 is chosen independently and uniformly at random, we get the following recursion:

$$\mathbb{E}\left[\mathcal{T}(k)\right]\le\sum_{j=1}^{|\mathcal{N}(k)|}\frac{j}{|\mathcal{N}(k)|}\cdot\mathbb{E}\left[\mathcal{T}(k-1)\right]=\frac{|\mathcal{N}(k)|+1}{2}\,\mathbb{E}\left[\mathcal{T}(k-1)\right].$$

Therefore, the total number of recursive calls is upper bounded by:

$$\mathbb{E}\left[\mathcal{T}(n_1+n_2)\right]\le\frac{\Pi_{i\in S_{\min}\cup S_{\max}}(|\mathcal{N}(i)|+1)}{2^{n_1+n_2}}.$$

□

Remark 7. *In Items 1 and 2, we know that while classic algorithms like* PI *and* VI *diverge, the less studied algorithm of* SPI *converges. We also remark that due to a lower bound in [13] for halting MDPs,* SPI *in the worst case has to go through all possible choices of neighbors and makes exponentially many iterations.*

In Item 3 of Theorem 6, the convergence rate of RandSPI *shows that it is in expectation exponentially faster than the brute force that naively enumerates all* $\Pi_{i\in S_{\min}\cup S_{\max}}|\mathcal{N}(i)|$ *possible neighbor choices of min and max variables.*

5 Discussions

Since all the problem classes {C1}, {C1+}, and {C1,C2} are in the class of UP ∩ coUP, any strict separation between these classes would imply P ≠ UP ∩ coUP and is, thus, notoriously difficult to prove. However, the algorithmic studies in this paper reveal that the classic algorithms (like policy iteration and value iteration) behave very differently for these problem classes, which indicates that they may have intrinsically different structures.

As one of the key results of this paper, we know that for LEMMs under {C1+}, value iteration is a polynomital time algorithm when γ is bounded away from 1. It is an interesting open question whether there exists a polynomial time algorithm for LEMMs under {C1} (provided γ bounded away from 1).

It is also known that a randomized subexponential-time algorithm exists for SSGs [12] and, thus, for the LEMMs under {C1,C2} following from the linear reduction (cf. Lemma 1). However, we do not know whether this randomized subexponential upper bound can be extended to the LEMMs under {C1} or under {C1+}. The problem is that the analysis from Ludwig [12] relies heavily on monotonicity, which does not hold anymore due to the negative coefficients. Extending the subexponential upper bound or proving any exponential lower bound for LEMMs under {C1} or under {C1+} is an interesting future direction.

Given an absolutely halting LEMM, it is not clear how to estimate the new conditioning γ efficiently. This is indeed a very non-trivial problem for the field of spectral analysis. Since the spectral radius can be irrational, most of the existing numerical schemes are not in polynomial time, not even for the fundamental classes of Markov chains. We propose a simple binary search method in Section D, and leave the more thorough numerical analysis as a fruitful future direction.

Acknowledgments

This work is partially supported by the French Agence Nationale de la Recherche (ANR) ANR-21-CE40-0020, the Austrian Science Fund (FWF) 10.55776/COE12, and the ERC CoG 863818 (ForM-SMArt) grant.

References

[1] Daniel Andersson and Peter Bro Miltersen. 2009. The complexity of solving stochastic games on graphs. In *International Symposium on Algorithms and Computation*. Springer, 112–121.

[2] Sally C Brailsford, Chris N Potts, and Barbara M Smith. 1999. Constraint satisfaction problems: Algorithms and applications. *European journal of operational research* 119, 3 (1999), 557–581.

[3] Krishnendu Chatterjee and Nathanaël Fijalkow. 2011. A reduction from parity games to simple stochastic games. In *Proceedings of the Second International Symposium on Games, Automata, Logics and Formal Verification (GandALF) (EPTCS, Vol. 54)*. 74–86. https://doi.org/10.4204/EPTCS.54.6

[4] Krishnendu Chatterjee, Ruichen Luo, Raimundo Saona, and Jakub Svoboda. 2025. Linear Equations with Min and Max Operators: Computational Complexity. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 11150–11157.

[5] Krishnendu Chatterjee, Tobias Meggendorfer, Raimundo Saona, and Jakub Svoboda. 2023. Faster algorithm for turn-based stochastic games with bounded treewidth. In *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 4590–4605. https://doi.org/10.1137/1.9781611977554.ch173

[6] Anne Condon. 1990. On Algorithms for Simple Stochastic Games. *Advances in computational complexity theory* 13 (1990), 51–72. https://doi.org/10.1090/dimacs/013/04

[7] Anne Condon. 1992. The Complexity of Stochastic Games. *Information and Computation* 96, 2 (1992), 203–224. https://doi.org/10.1016/0890-5401(92)90048-K

[8] Thomas Dueholm Hansen, Peter Bro Miltersen, and Uri Zwick. 2013. Strategy iteration is strongly polynomial for 2-player turn-based stochastic games with a constant discount factor. *Journal of the ACM (JACM)* 60, 1 (2013), 1–16.

[9] Guy Katz, Clark Barrett, David L Dill, Kyle Julian, and Mykel J Kochenderfer. 2017. Reluplex: An efficient SMT solver for verifying deep neural networks. In *Proceedings of the 29th International Conference on Computer Aided Verification (CAV)*. 97–117.

[10] David A Levin and Yuval Peres. 2017. *Markov chains and mixing times*. Vol. 107. American Mathematical Soc.

[11] Michael Lederman Littman. 1996. *Algorithms for sequential decision-making*. Brown University.

[12] Walter Ludwig. 1995. A subexponential randomized algorithm for the simple stochastic game problem. *Information and computation* 117, 1 (1995), 151–155. https://doi.org/10.1006/inco.1995.1035

[13] Mary Melekopoglou and Anne Condon. 1990. *On the complexity of the policy iteration algorithm*. University of Wisconsin-Madison, Computer Sciences Department.

[14] Martin L. Puterman. 2014. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley.

[15] Singiresu S Rao, R Chandrasekaran, and KPK Nair. 1973. Algorithms for discounted stochastic games. *Journal of Optimization Theory and Applications* 11, 6 (1973), 627–637.

[16] Bruno Scherrer. 2016. Improved and Generalized Upper Bounds on the Complexity of Policy Iteration. *Mathematics of Operations Research* 41, 3 (2016), 758–774. https://doi.org/10.1287/moor.2015.0753

[17] Shizuo Senju and Yoshiaki Toyoda. 1968. An approach to linear programming with 0-1 variables. *Management Science* (1968), B196–B207.

[18] L. S. Shapley. 1953. Stochastic Games. *Proceedings of the National Academy of Sciences* 39, 10 (1953), 1095–1100. https://doi.org/10.1073/pnas.39.10.1095

[19] Karl Sigmund and Martin A Nowak. 1999. Evolutionary game theory. *Current Biology* 9, 14 (1999), R503–R505.

[20] Yinyu Ye. 2011. The simplex and policy-iteration methods are strongly polynomial for the Markov decision problem with a fixed discount rate. *Mathematics of Operations Research* 36, 4 (2011), 593–603.

A Checking halting condition is coNP-complete

It is proven in [4] that checking Condition C1 is coNP-hard. In order to show that checking Condition C1 is coNP-complete, it sufficies to prove its coNP membership.

Assume Condition C1 does not hold. Let $\mathbf{Q}^{(a)} \in \mathcal{Q}$. If $\varrho(\mathbf{Q}^{(a)}) \geq 1$, we already have a polynomial certificate $\mathbf{Q}^{(a)}$. Now, let us assume $\varrho(\mathbf{Q}^{(a)}) < 1$. Since Condition C1 does not hold, there exists a matrix in $\mathrm{conv}(\mathcal{Q})$ such that it has an eigenvalue 1. Thus, there exists a matrix $\mathbf{Q} \in \mathrm{conv}\,\mathcal{Q}$ such that $(\mathbf{I} - \mathbf{Q})$ is a singular matrix–that is, a linear combination of $(\mathbf{I} - \mathbf{Q})$'s rows equals $\mathbf{0}$. Therefore, one can use the signs of the coefficients in the linear combination as a certificate, which can be checked by linear program in polynomial time.

B Missing proof

Proof of Theorem 1. (1) From Lemma 1, an LEMM under {C1+} can be reduced to an LEMM under {C1+,C3}. Now, we show that an LEMM under {C1+,C3} can be further reduced to an LEMM under {C1+,C3,C4}, which generally follows the reduction in [4, Appendix A.4].

Consider an LEMM with $(S_{\min}, S_{\max}, S_{\mathrm{aff}}, n, \mathcal{N}, \mathbf{q}, \mathbf{b})$ under Conditions C1+ and C3. Define, for $i \in [1, n_1]$,

$$\mathcal{N}'(i) = \{l + 2n \mid l \in \mathcal{N}(i) \cap ([1, n_1] \cup [n_1 + n_2 + 1, n])\} \cup \{l + n \mid l \in \mathcal{N}(i) \cap [n_1 + 1, n_1 + n_2]\},$$

for $i \in [n_1 + 1, n_1 + n_2]$,

$$\mathcal{N}'(i) = \{l + n \mid l \in \mathcal{N}(i) \cap ([1, n_1] \cup [n_1 + n_2 + 1, n])\} \cup \{l + 2n \mid l \in \mathcal{N}(i) \cap [n_1 + 1, n_1 + n_2]\},$$

and for $k \in [n_2 + 1, n]$,

$$\tilde{\mathbf{q}}_k = \begin{bmatrix} \mathbf{O}_{(n+n_1)\times n_1} & \mathbf{O}_{(n+n_1)\times n_2} & \mathbf{O}_{(n+n_1)\times(n-n_1-n_2)} \\ \mathbf{O}_{n_2\times n_1} & \mathbf{I}_{n_2} & \mathbf{O}_{n_2\times(n-n_1-n_2)} \\ \mathbf{O}_{(n-n_1-n_2)\times n_1} & \mathbf{O}_{(n-n_1-n_2)\times n_2} & \mathbf{O}_{(n-n_1-n_2)\times(n-n_1-n_2)} \\ \mathbf{I}_{n_1} & \mathbf{O}_{n_1\times n_2} & \mathbf{O}_{n_1\times(n-n_1-n_2)} \\ \mathbf{O}_{n_2\times n_1} & \mathbf{O}_{n_2\times n_2} & \mathbf{O}_{n_2\times(n-n_1-n_2)} \\ \mathbf{O}_{(n-n_1-n_2)\times n_1} & \mathbf{O}_{(n-n_1-n_2)\times n_2} & \mathbf{I}_{n-n_1-n_2} \end{bmatrix} \cdot \mathbf{q}_k\,.$$

We consider the following LEMM:

$$\begin{cases} x'_i = \min\{x'_l \mid l \in \mathcal{N}'(i)\}, & 1 \leq i \leq n_1 + n_2, \\ x'_k = \tilde{\mathbf{q}}_k^\top \mathbf{x}' + b_k, & n_1 + n_2 < k \leq n, \\ x'_l = -x'_{l-n}, & n < l \leq 2n, \\ x'_m = x'_{m-2n}, & 2n < m \leq 3n, \end{cases} \tag{18}$$

which satisfies Conditions C1+, C3 and C4. Then, $\mathbf{x}$ is a solution to the original LEMM if and only if $\mathbf{x}' = \begin{bmatrix} \tilde{\mathbf{x}} \\ -\tilde{\mathbf{x}} \\ \tilde{\mathbf{x}} \end{bmatrix}$ is the solution to (18), where

$$\tilde{\mathbf{x}} = \begin{bmatrix} \mathbf{I}_{n_1} & \mathbf{O}_{n_1\times n_2} & \mathbf{O}_{n_1\times(n-n_1-n_2)} \\ \mathbf{O}_{n_2\times n_1} & -\mathbf{I}_{n_2} & \mathbf{O}_{n_2\times(n-n_1-n_2)} \\ \mathbf{O}_{(n-n_1-n_2)\times n_1} & \mathbf{O}_{(n-n_1-n_2)\times n_2} & \mathbf{I}_{n-n_1-n_2} \end{bmatrix} \cdot \mathbf{x}\,.$$

Therefore, the LEMM under {C1+,C3,C4} is as hard as the LEMM under {C1+,C3}.

(2) Lemma 1 shows that the LEMMs under {C1,C2} and {C1,C2,C3} are equivalent. Then, the claim follows immediately from [7, Lemma 8], which shows the equivalence between the LEMMs under {C2,C3} and {C1,C2,C3}. □

C Stochastic games with discounted-sum objectives

Consider a two-player turn-based zero-sum stochastic games with discounted-sum objectives where the discount factor $\gamma < 1$. According to the classic results [8, 15, 18], the game can be formualted as an LEMM with $(S_{\min}, S_{\max}, S_{\mathrm{aff}}, n, \mathcal{N}, \mathbf{q}, \mathbf{b})$, where the following properties hold: (i) Conditions C2 and C3 are satisfied; (ii) $\mathcal{N}(i) \subseteq S_{\mathrm{aff}}$, for all $i \in S_{\min} \cup S_{\max}$; and (iii) $\mathbf{q}_k^\top \mathbf{1} = \gamma$, for all $k \in S_{\mathrm{aff}}$.

Due to the properties (i) and (iii), Condition C1 is trivially satisfied. Therefore, Corollary 5 follows directly from Theorem 4.

D A feasible method to estimate γ

We propose a simple binary search method to estimate the new halting conditioning γ introduced in Eq. (9).

Let us consider an absolutely halting LEMM. Let $\mathrm{LP}(\hat{\gamma})$ denote the following system of linear inequalities:

$$\begin{cases} \hat{\gamma}\mathbf{v} \geq |\mathbf{Q}|\mathbf{v} + \mathbf{1}, & \text{for all } \mathbf{Q} \in \mathcal{Q}, \\ \mathbf{v} \geq 1. \end{cases}$$

Let $\underline{\gamma}^{(0)} = 0$ and $\overline{\gamma}^{(0)} = \tilde{\gamma}$. For all $t \geq 0$, let $\hat{\gamma}^{(t)} = \frac{1}{2}(\underline{\gamma}^{(t)} + \overline{\gamma}^{(t)})$, and let

$$\underline{\gamma}^{(t+1)} = \begin{cases} \underline{\gamma}^{(t)}, & \text{if } \mathrm{LP}(\hat{\gamma}^{(t)}) \text{ is feasible}, \\ \hat{\gamma}^{(t)}, & \text{otherwise}, \end{cases}$$

and

$$\overline{\gamma}^{(t+1)} = \begin{cases} \overline{\gamma}^{(t)}, & \text{if } \mathrm{LP}(\hat{\gamma}^{(t)}) \text{ is not feasible}, \\ \hat{\gamma}^{(t)}, & \text{otherwise}. \end{cases}$$

Then, $\overline{\gamma}^{(t)} - \underline{\gamma}^{(t)} \leq 2^{-t}\tilde{\gamma}$. Moreover, in view of Proposition 4 and Item 4 in Lemma 2, we have

$$\gamma \in [\underline{\gamma}^{(t)}, \overline{\gamma}^{(t)}],$$

for all $t \geq 1$.