Simple Stochastic Games and P-matrix Generalized Linear Complementarity Problems*

Bernd Gärtner[1] and Leo Rüst[1]

Institute of Theoretical Computer Science, ETH Zürich, CH-8092 Zürich, Switzerland, {gaertner,ruestle}@inf.ethz.ch

Abstract. We show that the problem of finding optimal strategies for both players in a simple stochastic game reduces to the generalized linear complementarity problem (GLCP) with a P-matrix, a well-studied problem whose hardness would imply NP = co-NP. This makes the rich GLCP theory and numerous existing algorithms available for simple stochastic games. As a special case, we get a reduction from binary simple stochastic games to the P-matrix linear complementarity problem (LCP).

1 Introduction

Simple stochastic games (SSG) form a subclass of general stochastic games, introduced by Shapley in 1953 [1]. SSG are two-player games on directed graphs, with certain random moves. If both players play optimally, their respective strategies assign values $v(i)$ to the vertices i, with the property that the first player wins with probability $v(i)$, given the game starts at vertex i. For a given start vertex s, the optimization problem associated with the SSG is to compute the *game value* $v(s)$; the decision problem asks whether the game value is at least $1/2$.

Previous work. Condon was first to study the complexity-theoretic aspects of SSG [2]. She showed that the decision problem is in NP ∩ co-NP. This is considered as evidence that the problem is not NP-complete, because the existence of an NP-complete problem in NP ∩ co-NP would imply NP = co-NP. Despite this evidence and a lot of research, the question whether a polynomial time algorithm exists remains open.

SSG are significant because they allow polynomial-time reductions from other interesting classes of games. Zwick and Paterson proved a reduction from *mean payoff games* [3] which in turn admit a reduction from *parity games*, a result of Puri [4].

In her survey article from 1992, Condon reviews a number of algorithms for the optimization problem (and shows some of them to be incorrect) [5]. These algorithms compute optimal strategies for both players (we will say that they

* The authors acknowledge support from the Swiss Science Foundation (SNF), Project No. 200021-100316/1.

solve the game). For none of these algorithms, the (expected) worst-case behavior is known to be better than exponential in the number of graph vertices.

Ludwig was first to show that simple stochastic games can be solved in *subexponential* time [6], in the *binary* case where all outdegrees of the underlying graph are two. Under the known polynomial-time reduction from the general case to the binary case [3], Ludwig's algorithm becomes exponential, though. Björklund et al. [7] and independently Halman [8] established a subexponential algorithm also in the general case.

These subexponential methods had originally been developed for *linear programming* and the more general class of *LP-type problems*, independently by Kalai [9, 10] as well as Matoušek, Sharir and Welzl [11]. Ludwig's contribution was to extract the combinatorial structure underlying binary SSG, and to show that this structure allows the subexponential algorithms to be applied. Halman was first to show that the problem of finding an optimal strategy for one of the players can actually be formulated as an LP-type problem [12]. Given a strategy, the other player's best response can be computed by a linear program. In a later result, Halman avoided linear programming by computing the other player's best response again by an LP-type algorithm. This resulted in *strongly* subexponential algorithms, the best known to date [8].

Independently, Björklund et al. arrived at subexponential methods by showing that SSG (as well as mean payoff and parity games) give rise to very specific LP-type problems [7]. Their contribution was to map all three classes of games to the single combinatorial problem of optimizing a *completely local-global* function over the Cartesian product of sets. Along with this, they also carried out an extensive study concerning the combinatorial properties of such functions.

Our contribution. In this paper, we show that the problem of solving a general (not necessarily binary) simple stochastic game can be written as a *generalized linear complementarity problem* (GLCP) with a P-matrix. The GLCP, as introduced by Cottle and Dantzig [13], consists of a *vertical block* $(m \times n)$*-matrix* M where $m \geq n$ and a right-hand side m-vector q. M and q are partitioned in conformity into n horizontal blocks M^i and q^i, $i = 1, \ldots n$, where the size of block i is $m_i \times n$ in M and m_i in q. Solving a GLCP means to find a nonnegative m-vector w partitioned in conformity with M and q and a nonnegative n-vector z such that

$$\begin{aligned} & w - Mz = q, && (1) \\ & \prod_{j=1}^{m_i} w_j^i z_i = 0, \quad \forall i \in \{1, \ldots, n\}. \end{aligned}$$

Here and in the following, w_j^i is the j-th element in the i-th block of w. M_j^i will denote the j-th row of the i-th block of M. A *representative submatrix* of M is an $(n \times n)$-matrix whose i-th row is M_j^i for some $j \in \{1, \ldots, m_i\}$. M is defined to be a P-matrix, if all principal minors of all its representative submatrices are positive [13].

In this paper, we will consider SSG with vertices of arbitrary outdegree and with *average* vertices determining the next vertex according to an arbitrary probability distribution. This is a natural generalization of binary SSG introduced by Condon [2]. A binary SSG will reduce to the more popular *linear complementarity problem* (LCP) where $m_i = 1$ for all n blocks.

The fact that there is a connection between games and LCP is not entirely surprising, as for example bimatrix games can be formulated as LCP [14]. Also Cottle, Pang and Stone [15, Section 1.2] list a simple game on Markov chains as an application for LCP, and certain (very easy) SSG are actually of the type considered.

LCP and methods for solving them are well-studied for general matrices M, and for specific matrix classes. The book by Cottle, Pang and Stone is the most comprehensive source for the rich theory of LCP, and for the various algorithms that have been developed to solve general and specific LCP [15]. A lot of results carry over to the GLCP. The significance of the class of P-matrices comes from the fact that M is a P-matrix if and only if the GLCP has a unique solution (w, z) for *any* right-hand side q [16]. Given this, the fact that our reduction yields a P-matrix already follows from Shapley's results. His class of games contains a superclass of SSG for which our reduction may yield *any* right-hand side q in (1). Shapley's theorem proving uniqueness of *game values* then implies that the matrix M in (1) must be a P-matrix. Our result provides an alternative proof of Shapley's theorem, specialized to SSG, and it makes the connection to matrix theory explicit.

No polynomial-time methods are known to solve P-matrix LCP, but Megiddo has shown that NP-hardness of the problem implies NP = co-NP [17], meaning that the problem has an unresolved complexity status, similar to that of SSG. Megiddo's proof easily carries over to P-matrix GLCP.

Gärtner et al. proved that the combinatorial structure of P-matrix GLCP is very similar to the structure derived by Björklund et al. for the games [18]. The latter authors also describe a reduction to what they call *controlled linear programming* [19]; controlled linear programs are easily mapped to (non-standard) LCP. Independently from our work, Björklund et al. have made this mapping explicit by deriving LCP-formulations for mean payoff games [20]. Their reduction is very similar to ours, but the authors do not prove that the resulting matrices are P-matrices, or belong to some other known class. In fact, Björklund et al. point out that the matrices they get are in general not P-matrices, and this stops them from further investigating the issue. We have a similar phenomenon here: Applying our reduction to *non-stopping* SSG (see next section), we may also obtain matrices that are not P-matrices. The fact that comes to our rescue is that the stopping assumption incurs no loss of generality. It would be interesting to see whether the matrices of Björklund et al. are actually P-matrices as well, after some transformation applied to the mean payoff game.

Matrix classes and algorithms. Various solution methods have been devised for GLCP, and when we specialize to the P-matrix GLCP we get from SSG, some of them are already familiar to the game community. Most notably, *principal pivot* *algorithms* in the GLCP world correspond to *switching* algorithms. Such an algorithm maintains a pair of strategies for both players and gradually improves them by locally switching to a different behavior. There exist examples of SSG where switching algorithms may cycle [5]. However, if switching is defined with respect to only one player (where after each switch the optimal counterstrategy of the other player is recomputed by solving a linear program), cycling is not possible. The latter is the setup of Björklund et al. [7].

In order to assess the power of the GLCP approach, we must understand the class of matrices resulting from SSG. Our result that these are P-matrices puts SSG into the realm of 'well-behaved' GLCP, but it does not give improved runtime bounds, let alone a polynomial-time algorithm. The major open question resulting from our approach is therefore the following: Can we characterize the subclass of P-matrices resulting from SSG? Is this subclass equal (or related) to some known class? In order to factor out the peculiarities of our reduction, we should require the subclass to be closed under scaling of rows and/or columns. Without having a concrete example, we believe that we obtain a proper subclass of the class of all P-matrices. This is because the GLCP restricted to (the variables coming from) any one of the two players is easy to solve by linear programming, a phenomenon that will not occur for a generic P-matrix.

The class of *hidden* K-*matrices* is one interesting subclass of P-matrices for which the GLCP can be solved in polynomial time. Mohan and Neogy [21] have generalized results by Pang [22] and Mangasarian [23, 24] to show that a vertical block hidden K-matrix can be recognized in polynomial time through a linear program and that the solution to this linear program can in turn be used to set up another linear program for solving the GLCP itself.

The matrices we get from SSG are in general not hidden K-matrices. Still, properties of the subclass of matrices we ask for might allow their GLCP to be solved in polynomial time.

2 Simple Stochastic Games

We are given a finite directed graph G whose vertex set has the form

$$V = \{\mathbf{1}, \mathbf{0}\} \cup V_{max} \cup V_{min} \cup V_{avg},$$

where $\mathbf{1}$, the *1-sink*, and $\mathbf{0}$, the *0-sink*, are the only two vertices with no outgoing edges. For reasons that become clear later, we allow multiple edges in G (in which case G is actually a multigraph).

Vertices in V_{max} belong to the first player which we call the *max player*, while vertices in V_{min} are owned by the second player, the *min player*. Vertices in V_{avg} are *average* vertices. For $i \in V \setminus \{\mathbf{1}, \mathbf{0}\}$, we let $\mathcal{N}(i)$ be the set of neighbors of i along the outgoing edges of i. The elements of $\mathcal{N}(i)$ are $\{\eta_1(i), \ldots, \eta_{|\mathcal{N}(i)|}(i)\}$. An average vertex i is associated with a probability distribution $\mathcal{P}(i)$ that assigns to each outgoing edge (i, j) of i a probability $p_{ij} > 0$, $\sum_{j \in \mathcal{N}(i)} p_{ij} = 1$.

The SSG defined by G is played by moving a token from vertex to vertex, until it reaches either the 1-sink or the 0-sink. If the token is at vertex i, it is moved according to the following rules.

vertex type	rule
$i = \mathbf{1}$	the game is over and the max player wins
$i = \mathbf{0}$	the game is over and the min player wins
$i \in V_{max}$	the max player moves the token to a vertex in $\mathcal{N}(i)$
$i \in V_{min}$	the min player moves the token to a vertex in $\mathcal{N}(i)$
$i \in V_{avg}$	the token moves to a vertex in $\mathcal{N}(i)$ according to $\mathcal{P}(i)$

An SSG is called *stopping*, if no matter what the players do, the token eventually reaches **1** or **0** with probability 1, starting from *any* vertex. In a stopping game, there are no directed cycles involving only vertices in $V_{max} \cup V_{min}$. The following is well-known and has first been proved by Shapley [1], see also the papers by Condon [2, 5]. Our reduction yields an independent proof of part (i).

Lemma 1. *Let G define a stopping* SSG.

(i) There are unique numbers $v(i), i \in G$, satisfying the equations

$$v(i) = \begin{cases} 1, & i = \mathbf{1} \\ 0, & i = \mathbf{0} \\ \max\limits_{j \in \mathcal{N}(i)} (v(j)), & i \in V_{max} \\ \min\limits_{j \in \mathcal{N}(i)} (v(j)), & i \in V_{min} \\ \sum\limits_{j \in \mathcal{N}(i)} p_{ij} v(j), & i \in V_{avg} \end{cases} . \quad (2)$$

(ii) The value $v(i)$ is the probability for reaching the 1-sink from vertex i, if both players play optimally.

For a discussion about what it means that 'both players play optimally', we refer to Condon's paper [5]. The important point here is that computing the numbers $v(i)$ solves the optimization version of the SSG in the sense that for every possible start vertex s, we know the *value* $v(s)$ of the game. It also solves the decision version which asks whether $v(s) \geq 1/2$. Additionally, the lemma shows that there are *pure* optimal strategies that can be read off the numbers $v(i)$: If v is a solution to (2), then an optimal pure strategy is given by moving from vertex i along one outoing edge to a vertex j with $v(j) = v(i)$.

The stopping assumption can be made without loss of generality: In a non-stopping game, replace every edge (i, j) by a new average vertex t_{ij} and new edges (i, t_{ij}) (with the same probability as (i, j) if $i \in V_{avg}$), (t_{ij}, j) with probability $1 - \epsilon$ and $(t_{ij}, \mathbf{0})$ with probability ϵ. Optimal strategies to this stopping game (which are given by the $v(i)$ values) correspond to optimal strategies in the original game if ϵ is chosen small enough [2].

3 Reduction to P-matrix GLCP

In the following, we silently assume that G defines a stopping SSG and that every non-sink vertex of G has at least two outgoing edges (a vertex of outdegree 1 can be removed from the game without affecting the values of other vertices). In order to solve (2), we first write down an equivalent system of linear equations and inequalities, along with (nonlinear) *complementarity* conditions for certain pairs of variables. The system has one variable x_i for each vertex i and one *slack variable* y_{ij} for each edge (i,j) with $i \in V_{max} \cup V_{min}$. It has equality constraints

$$x_i = \begin{cases} 1, & i = \mathbf{1} \\ 0, & i = \mathbf{0} \\ y_{ij} + x_j, & i \in V_{max},\, j \in \mathcal{N}(i) \\ -y_{ij} + x_j, & i \in V_{min},\, j \in \mathcal{N}(i) \\ \displaystyle\sum_{j \in \mathcal{N}(i)} p_{ij} x_j, & i \in V_{avg}, \end{cases} \tag{3}$$

inequality constraints

$$y_{ij} \geq 0, \quad i \in V_{max} \cup V_{min},\, j \in \mathcal{N}(i), \tag{4}$$

and complementarity constraints

$$\prod_{j \in \mathcal{N}(i)} y_{ij} = 0, \quad i \in V_{max} \cup V_{min} \tag{5}$$

to model the max- and min-behavior in (2).

The statement of Lemma 1 (i) is equivalent to the statement that the system consisting of (3), (4) and (5) has a unique solution $x = (x_1, \ldots, x_n)$, and we will prove the latter statement. From the solution x, we can recover the game values via $v(i) = x_i$, and we also get the y_{ij}. Note that edges with $y_{ij} = 0$ in the solution correspond to *strategy edges* of the players.

It turns out that the variables x_i are redundant, and in order to obtain a proper GLCP formulation, we will remove them. For variables $x_i, i \notin V_{avg}$, this is easy.

Definition 1. *Fix $i \in V$.*

(i) *The* first path *of i is the unique directed path that starts from i, consists only of edges $(j, \eta_1(j))$ with $j \in V_{max} \cup V_{min}$, and ends at some vertex in $\{\mathbf{1}, \mathbf{0}\} \cup V_{avg}$. The* second path *is defined analogously, with edges of the form $(j, \eta_2(j))$.*

(ii) *The* substitution *S_i of x_i is recursively defined as the linear polynomial*

$$S_i = \begin{cases} 1, & i = \mathbf{1} \\ 0, & i = \mathbf{0} \\ y_{i\eta_1(i)} + S_{\eta_1(i)}, & i \in V_{max} \\ -y_{i\eta_1(i)} + S_{\eta_1(i)}, & i \in V_{min} \\ x_i, & i \in V_{avg} \end{cases}. \tag{6}$$

(iii) *$\bar{S}_i$ is the homogeneous polynomial obtained from S_i by removing the constant term (which is 0 or 1, as a consequence of (ii)).*

Note that the first and the second path always exist, because there are no cycles involving only vertices in $V_{max} \cup V_{min}$ and each non-sink vertex has outdegree at least 2. The substitution S_i expresses x_i in terms of the y-variables associated with the first path edges of i, and in terms of the substitution of the last vertex on the first path, which is either an average vertex, or a sink.

Lemma 2. *The following system of equations is equivalent to (3).*

$$\begin{array}{rll} y_{ij} = y_{i\eta_1(i)} + S_{\eta_1(i)} - S_j, & i \in V_{max},\, j \in \mathcal{N}(i) \setminus \eta_1(i) \\ y_{ij} = y_{i\eta_1(i)} - S_{\eta_1(i)} + S_j, & i \in V_{min},\, j \in \mathcal{N}(i) \setminus \eta_1(i) \\ 0 = x_i - \displaystyle\sum_{j \in \mathcal{N}(i)} p_{ij} S_j, & i \in V_{avg}. \end{array} \tag{7}$$

Proof. By induction on the length of the first path, it can be shown that in every feasible solution to (3), x_i has the same value as its substitution. The system (3) therefore implies (7). Vice versa, given any solution to (7), we can simply set

$$x_i = \begin{cases} 1, & i = \mathbf{1} \\ 0, & i = \mathbf{0} \\ y_{i\eta_1(i)} + S_{\eta_1(i)}, & i \in V_{max} \\ -y_{i\eta_1(i)} + S_{\eta_1(i)}, & i \in V_{min} \end{cases}$$

to guarantee that x_i and S_i have the same value for all i. Then, the equations of (7) imply that x_i satisfies (3), for all i. □

3.1 A non-standard GLCP

Let us assume that $V_{max} \cup V_{min} = \{1, \dots, u\}, V_{avg} = \{u+1, \dots, n\}$. Moreover, for $i, j \in \{1, \dots, u\}$ and $i < j$, we assume that there is no directed path from j to i that avoids average vertices. This is possible, because G restricted to $V \setminus V_{avg}$ is acyclic, by our stopping assumption. In other words, the order $1, \dots, u$ topologically sorts the vertices in $V_{max} \cup V_{min}$, with respect to the subgraph induced by $V \setminus V_{avg}$. Defining vectors

$$z = (y_{1\eta_1(1)}, \dots, y_{u\eta_1(u)})^T, \quad w = (w^1, \dots, w^u)^T, \quad x = (x_{u+1}, \dots, x_n)^T, \tag{8}$$

where $w^i = (y_{i\eta_2(i)}, \dots, y_{i\eta_{|\mathcal{N}(i)|}(i)})$ is the vector consisting of the y_{ij} for all $j \in \mathcal{N}(i) \setminus \eta_1(i)$, conditions (4), (5) and (7)— and therefore the problem of computing the $v(i)$—can now be written as

$$\begin{array}{ll} \text{find} & w, z \\ \text{subject to} & w \geq 0, z \geq 0, \\ & \displaystyle\prod_{j=1}^{|\mathcal{N}(i)|-1} w^i_j z_i = 0, \quad i \in V_{max} \cup V_{min} \\ & \begin{pmatrix} w \\ 0 \end{pmatrix} - \begin{pmatrix} Q & C \\ A & B \end{pmatrix} \begin{pmatrix} z \\ x \end{pmatrix} = \begin{pmatrix} s \\ t \end{pmatrix}, \end{array} \tag{9}$$

where

$$P = \begin{pmatrix} Q & C \\ A & B \end{pmatrix}, \quad q = \begin{pmatrix} s \\ t \end{pmatrix}$$

are a suitable matrix and a suitable vector. The vertical block matrix Q is partitioned according to w and encodes the connections between player vertices (along first paths), whereas the square matrix B encodes the connections between average vertices. A and C describe how player and average vertices interconnect.

3.2 The structure of the matrix *P*

The following three lemmas are needed to show that P is a P-matrix, i.e. all principal minors of all representative submatrices of P are positive.

Lemma 3. *Q is a* P*-matrix.*

Proof. Every representative submatrix of Q is upper-triangular, with all diagonal entries being equal to 1. The latter fact is a direct consequence of (7), and the former follows from our topological sorting: For any $k \in \mathcal{N}(j)\setminus\eta_1(j)$, the variable $y_{i\eta_1(i)}$ cannot occur in the equation of (7) for y_{jk}, $j > i$, because this would mean that i is on the first path of either $\eta_1(j) > j > i$ or $k > j > i$. Thus, every representative submatrix has determinant 1. □

It can even be shown that Q is a hidden K-matrix [25].

Lemma 4. *B is a* P*-matrix.*

Proof. We may assume that the average vertices are 'topologically sorted' in the following sense. For $i, j \in \{u+1, \ldots, n\}$ and $i \leq j$, there is a neighbor $k \in \mathcal{N}(i)$ such that the first path of k avoids j. To construct this order, we use our stopping assumption again. Assume we have built a prefix of the order. Starting the game in one of the remaining average vertices, and with both players always moving the token along edges $(i, \eta_1(i))$, we eventually reach a sink. The last of the remaining average vertices on this path is the next vertex in our order.

Theorem 3.11.10 in the book by Cottle, Pang and Stone [15] states that a square matrix M with all off-diagonal entries nonpositive is a P-matrix if there exists a positive vector s such that $Ms > 0$. As B has all off-diagonal entries nonpositive by construction (last line of (7)), it thus remains to provide the vector s. We define s to be the monotone increasing vector

$$s_t = 1 - \epsilon^t, \quad t = 1, \ldots, |V_{avg}|,$$

where $\epsilon = \min_{i,j} p_{ij}$ is the smallest probability occuring in the average vertices' probability distributions over the outgoing edges. $s > 0$ as $0 < \epsilon \leq 1/2$, and we claim that also $Bs > 0$. Consider the row of B corresponding to average vertex i, see (7). Its diagonal entry is a positive number x. We have $\epsilon \leq x \leq 1$ by our stopping assumption ($x < 1$ occurs if the first path of any neighbor of i comes back to i). The off-diagonal values of the row are all nonpositive and sum up to at least $-x$. Our topological sorting on the average vertices implies that the row elements to the right of x sum up to $-(x-\epsilon)$ at least. Assume that the diagonal element x is at position t in the row. Under these considerations, the value of the scalar product of the row with s is minimized if the last element of the row has value $-(x-\epsilon)$ and the element at position $t-1$ has value $-\epsilon$. As claimed, by the following formula the value is then positive (at least for $1 < t < |V_{avg}|$, but the cases $t = 1$ and $t = |V_{avg}|$ can be checked in the same way):

$$-\epsilon(1-\epsilon^{t-1}) + x(1-\epsilon^t) - (x-\epsilon)(1-\epsilon^{|V_{avg}|}) = \epsilon^t(1-x) + \epsilon^{|V_{avg}|}(x-\epsilon) > 0.$$

□

We note that B is even a K-matrix. K-matrices form a proper subclass of hidden K-matrices [15]

Property 1. An $(n \times n)$ representative submatrix

$$P_{rep} = \begin{pmatrix} Q_{rep} & C_{rep} \\ A & B \end{pmatrix}$$

of P is given by a representative submatrix Q_{rep} of Q and C_{rep} which consists of the rows of C corresonding to the rows of Q_{rep}. P_{rep} corresponds to a subgame where edges have been deleted such that every player vertex has exactly two outgoing edges.

Such a subgame is a slightly generalized binary SSG, as average vertices can have more than two outgoing edges and arbitrary probability distributions on them.

Lemma 5. *Using elementary row operations, we can transform a representative submatrix P_{rep} of P into a matrix P'_{rep} of the form*

$$P'_{rep} = \begin{pmatrix} Q_{rep} & C_{rep} \\ 0 & B' \end{pmatrix}$$

with B' being a P*-matrix.*

Proof. We process the rows of the lower part (AB) of P_{rep} one by one. In the following, first and second paths (and thus also substitutions $\bar{S}_i$) are defined w.r.t. the subgame corresponding to P_{rep}.

For $k \in \{1, \ldots, n\}$, let R_k be the k-th row of P_{rep} and assume that we are about to process $R_i, i \in \{u+1, \ldots, n\}$. According to (7), we have

$$R_i \begin{pmatrix} z \\ x \end{pmatrix} = x_i - \sum_{j \in \mathcal{N}(i)} p_{ij} \bar{S}_j. \tag{10}$$

We will eliminate the contribution of $\bar{S}_j$ for all $j \in \mathcal{N}(i)$, by adding suitable multiples of rows $R_k, k \in \{1, \ldots, u\}$. For such a k, (7) together with (6) implies

$$R_k \begin{pmatrix} z \\ x \end{pmatrix} = \begin{cases} \bar{S}_k - \bar{S}_{\eta_1(i)}, & k \in V_{max} \\ \bar{S}_{\eta_1(i)} - \bar{S}_k, & k \in V_{min} \end{cases}. \tag{11}$$

Let $\eta_2^*(j)$ be the last vertex on the second path of j. Summing up (11) over all vertices on the second path of vertex k with suitable multiples from $\{1, -1\}$, the sum telescopes, and we get that for all $j \in \{1, \ldots, u\}$, $\bar{S}_j - \bar{S}_{\eta_2^*(j)}$ is obtainable as a linear combination of the row vectors

$$R_k \begin{pmatrix} z \\ x \end{pmatrix}, \quad k \leq u.$$

Actually, if j is an average vertex or a sink, we have $\eta_2^*(j) = j$, so that $\bar{S}_j - \bar{S}_{\eta_2^*(j)} = 0$ is also obtainable as a (trivial) linear combination in this case.

Thus, adding $(\bar{S}_j - \bar{S}_{\eta_2^*(j)})p_{ij}$ to (10) for all $j \in \mathcal{N}(i)$ transforms our current matrix into a new matrix whose i-th row has changed and yields

$$R'_i \begin{pmatrix} z \\ x \end{pmatrix} = x_i - \sum_{j \in \mathcal{N}(i)} p_{ij} \bar{S}_{\eta_2^*(j)}. \tag{12}$$

Moreover, this transformation is realized through elementary row operations. Because (12) does not contain any y-variables anymore, we get the claimed structure after all rows $R_i, i \in \{u+1, \ldots, n\}$ have been processed.

We still need to show that B' is a P-matrix, but this is easy. B encodes for each average vertex the average vertices reached along the first paths of its successors. According to (12), B' does the same thing, but replacing first paths with second paths. The two situations are obviously completely symmetric, so the fact that B is a P-matrix also yields that B' is a P-matrix. Note that in order to obtain a monotone vector s as in the proof of Lemma 4, we need to reshuffle rows and columns so that the corresponding vertices are 'topologically sorted' according to second paths. □

Lemma 6. *P is a* P*-matrix.*

Proof. We show that every representative submatrix P_{rep} of P is a P-matrix. By Lemma 5, $\det(P_{rep}) = \det(P'_{rep}) = \det(Q_{rep})\det(B')$, so P_{rep} has positive determinant as both Q and B' are P-matrices by Lemmas 3 and 4. To see that all proper principal minors are positive, we can observe that any principal submatrix of P_{rep} is again the matrix resulting from a SSG. The subgame corresponding to a principal submatrix can be derived from the SSG by deleting vertices and redirecting edges. This may generate multiple edges, which is the reason why we allowed them in the definition of the SSG. (The easy details are omitted.) □

3.3 A standard GLCP

Problem (9) is a non-standard GLCP because there are variables x with no complementarity conditions. But knowing that B is regular (as B is a P-matrix), we can express x in terms of z and obtain an equivalent standard GLCP, whose matrix is a *Schur complement* of B in P.

$$
\begin{array}{ll}
\text{find } w, z & \\
\text{subject to} & w \geq 0, z \geq 0, \\
& w^T z = 0, \\
& w - (Q - CB^{-1}A)z = s - CB^{-1}t.
\end{array} \tag{13}
$$

Lemma 7. *$Q - CB^{-1}A$, is a* P*-matrix.*

Proof. We have to show that every representative submatrix of $Q - CB^{-1}A$ is a P-matrix. Such submatrices are derived through $Q_{rep} - C_{rep}B^{-1}A$. It thus suffices to show that $Q_{rep} - C_{rep}B^{-1}A$ is a P-matrix, given that P_{rep} (as defined in Property 1) is a P-matrix. This is well known (see for example Tsatsomeros [26]). □

We have finally derived our main theorem:

Theorem 1. *A simple stochastic game is reducible in polynomial time to a generalized linear complementarity problem with a* P*-matrix.*

This theorem also provides a proof of Lemma 1: going through our chain of reductions again yields that the equation system in Lemma 1 (i) for the values $v(i)$ has a unique solution if and only if the GLCP (13) has a unique solution for the slack variables y_{ij}. The latter holds because the matrix of (13) is a P-matrix.

As mentioned earlier, the reduction works for a superclass of SSGin which edges are associated with a payoff. But for general stochastic games as introduced by Shapley [1], the reduction (as described in this paper) is not possible. This follows from two facts. First, optimal strategies for stochastic games are generally non-pure. Second, it is possible to get irrational solutions (vertex values) for the stochastic game even if all input data is rational. This is not possible for GLCP.

Acknowledgments

We thank Walter Morris for providing us with many insights about matrix classes and LCP algorithms. We also thank Nir Halman for inspiring discussions.

References

1. Shapley, S.: Stochastic games. Proceedings of the National Academy of Sciences, U.S.A. **39** (1953) 1095–1100
2. Condon, A.: The complexity of stochastic games. Information & Computation **96** (1992) 203–224
3. Zwick, U., Paterson, M.: The complexity of mean payoff games on graphs. Theor. Comput. Sci. **158** (1996) 343–359
4. Puri, A.: Theory of hybrid systems and discrete event systems. PhD thesis, University of California at Berkeley (1995)
5. Condon, A.: On algorithms for simple stochastic games. In Cai, J.Y., ed.: Advances in Computational Complexity Theory. Volume 13 of DIMACS Series in Discrete Mathematics and Theoretical Computer Science. American Mathematical Society (1993) 51–73

6. Ludwig, W.: A subexponential randomized algorithm for the simple stochastic game problem. Information and Computation **117** (1995) 151–155
7. Björklund, H., Sandberg, S., Vorobyov, S.: Randomized subexponential algorithms for infinite games. Technical Report 2004-09, DIMACS: Center for Discrete Mathematics and Theoretical Computer Science, Rutgers University, NJ (2004)
8. Halman, N.: Discrete and Lexicographic Helly Theorems and Their Relations to LP-type problems. PhD thesis, Tel-Aviv University (2004)
9. Kalai, G.: A subexponential randomized simplex algorithm. In: Proc. 24th annu. ACM Symp. on Theory of Computing. (1992) 475–482
10. Kalai, G.: Linear programming, the simplex algorithm and simple polytopes. Math. Programming **79** (1997) 217–233
11. Matoušek, J., Sharir, M., Welzl, E.: A subexponential bound for linear programming. Algorithmica **16** (1996) 498–516
12. Halman, N.: An EGLP formulation for the simple stochastic game problem, or a comment on the paper: *A subexponential randomized algorithm for the Simple Stochastic Game problem* by W. Ludwig. Technical Report RP-SOR-01-02, Department of Statistics and Operations Research (2001)
13. Cottle, R.W., Dantzig, G.B.: A generalization of the linear complementarity problem. Journal on Combinatorial Theory **8** (1970) 79–90
14. von Stengel, B.: Computing equilibria for two-person games. In: Handbook of Game Theory. Volume 3. Elsevier Science Publishers (North-Holland) (2002)
15. Cottle, R.W., Pang, J., Stone, R.E.: The Linear Complementarity Problem. Academic Press (1992)
16. Szanc, B.P.: The Generalized Complementarity Problem. PhD thesis, Rensselaer Polytechnic Institute, Troy, NY (1989)
17. Megiddo, N.: A note on the complexity of P-matrix LCP and computing an equilibrium. Technical report, IBM Almaden Research Center, San Jose (1988)
18. Gärtner, B., Morris, Jr., W.D., Rüst, L.: Unique sink orientations of grids. In: Proc. 11th Conference on Integer Programming and Combinatorial Optimization (IPCO). Volume 3509 of Lecture Notes in Computer Science. (2005) 210–224
19. Björklund, H., Svensson, O., Vorobyov, S.: Controlled linear programming for infinite games. Technical Report 2005-13, DIMACS: Center for Discrete Mathematics and Theoretical Computer Science, Rutgers University, NJ (2005)
20. Björklund, H., Svensson, O., Vorobyov, S.: Linear complementarity algorithms for mean payoff games. Technical Report 2005-05, DIMACS: Center for Discrete Mathematics and Theoretical Computer Science, Rutgers University, NJ (2005)
21. Mohan, S.R., Neogy, S.K.: Vertical block hidden Z-matrices and the generalized linear complementarity problem. SIAM J. Matrix Anal. Appl. **18** (1997) 181–190
22. Pang, J.: On discovering hidden Z-matrices. In Coffman, C.V., Fix, G.J., eds.: Constructive Approaches to Mathematical Models. Proceedings of a conference in honor of R. J. Duffin, New York, Academic Press (1979) 231–241
23. Mangasarian, O.L.: Linear complementarity problems solvable by a single linear program. Math. Program. **10** (1976) 263–270
24. Mangasarian, O.L.: Generalized linear complementarity problems as linear programs. Oper. Res.-Verf. **31** (1979) 393–402
25. Gärtner, B., Rüst, L.: Properties of vertical block matrices. Manuscript (2005)
26. Tsatsomeros, M.J.: Principal pivot transforms: Properties and applications. Linear Algebra and Its Applications **307** (2000) 151–165