Generalized Amazons is PSPACE–Complete

Timothy Furtak[1], Masashi Kiyomi[2], Takeaki Uno[3], Michael Buro[4]

[1,4] Department of Computing Science, University of Alberta, Edmonton, Canada. email: {[1]furtak,[4]mburo}@cs.ualberta.ca

[2,3] National Institute of Informatics, Japan. email: [2]masashi@grad.nii.ac.jp, [3]uno@nii.jp

Abstract

Amazons is a perfect information board game with simple rules and large branching factors. Two players alternately move chess queen-like pieces and block squares on a 10×10 playing field. The player who makes the last move wins. Amazons endgames usually decompose into independent subgames. Therefore, the game is a natural testbed for combinatorial game theory. It was known that determining the winner of simple generalized Amazons endgames is NP-equivalent. This paper presents two proofs for the PSPACE-completeness of the generalized version of the full game.

1 Introduction

Amazons is a two-person perfect information board game that combines elements of chess and Go. Both players start with four pieces called amazons which are symmetrically positioned at the edges of an otherwise empty 10×10 square board. Moves consist of picking an amazon of the player's colour, moving it like a chess queen, and then shooting an arrow straight in one of eight directions from the amazon's destination square to an empty square. This square becomes blocked for the remainder of the game and no amazon or arrow can pass it. Arrows are not allowed to pass amazons either and amazons cannot be captured. Blocking squares is mandatory. The game proceeds in turns and the first player without any legal move loses. Fig. 1 shows two typical Amazons positions.

The game of Amazons is interesting for AI research for at least the following reasons:

- Compared with other popular games its rule set is simple — and yet, it leads to interesting strategic concepts which need to be addressed when designing evaluation functions [Lieberum, 2005].

- The number of legal moves in Amazons positions on the standard board is often in the hundreds or even thousands. This puts pressure on sound backward pruning search techniques such as the alpha-beta algorithm, because the number of positions to be searched even at small depths is astronomical. Amazons can therefore serve as a testbed for research on selective search and adversarial planning.

- Like Hex, Go-moku, Tic-Tac-Toe, and Othello, Amazons has the property of being monotonically increasing (each move blocks one square), so the underlying structure of the game is a directed acyclic graph, rather than a general graph. This eliminates some potentially complicating variables for scientific research. Also like Hex, Amazons cannot end in a draw.

- Endgames regularly decompose into independent subgames (Fig 1). At this stage, combinatorial game theory can be applied to vastly reduce the search space, and thus speed up good move selection compared with full-board analyses. Hence, Amazons is a suitable test-domain for new ideas in this area. See for instance [Berlekamp, 2000; Müller and Tegos, 2002; Snatzke, 2004].

Establishing the computational complexity of various problems associated with generalized games is also interesting. For example, proving NP- or PSPACE-hardness is convincing evidence that a problem does not possess a simple solution in general (although effective methods may exist for specific instances).
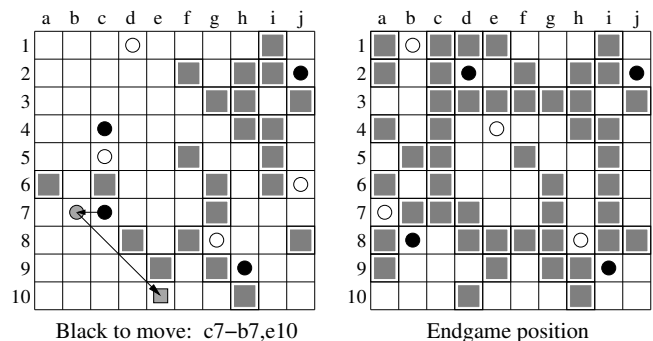
Generalizing the standard Amazons game is straight-

Figure 1: Typical Amazons positions. On the left, the black amazon at c7 is about to move to b7 and block e10 to prevent the white amazon at g8 from entering the SW corner area. On the right, Black can make 24 unopposed moves, whereas White can make only 20, so Black can win regardless of who is to move.

forward: we allow arbitrary square-shaped boards ($n \times n$) with at least one and up to n^2 amazons. It is known that solving generalized Amazons puzzles (i.e. determining whether in a single-player Amazons game the player can make at least k consecutive moves) is NP-complete and determining the winner of simple generalized Amazons endgames — in which players do not interact anymore — is NP-equivalent [Buro, 2000]. In what follows, we present two proofs for the PSPACE-hardness of deciding the winner of generalized[1] Amazons positions. While the first proof is more intuitive, the second one leads to a stronger results, namely that the problem remains PSPACE-complete even if each player has only one amazon.

2 Hex Reduction

In this section we show that determining the winner in Hex positions, which is known to be PSPACE-complete, can be reduced to the related Amazons problem. We begin with the definition of the sets we like to relate to each other:

Definition 1:

- **AMAZONS** is a set of succinct encodings of all generalized Amazons positions with a winning strategy for Black (to move).

- **HEX** is a set of succinct encodings of all Hex positions with a winning strategy for Black (to move).

Theorem 1: HEX $\leq_p$ AMAZONS.

Corollary 1: AMAZONS is PSPACE-complete.

Proof: [Reisch, 1981] shows that HEX is PSPACE-complete by reducing QBF to HEX. Moreover, AMAZONS $\in$ PSPACE, which is obvious by the existence of a depth-first minimax search algorithm that decides the winner of a given Amazons position in space polynomial in the length of its encoding. Note, that the number of moves in an Amazons game played on an $n \times n$ board is at most n^2. $\qquad\square$

Proof of Theorem 1: We describe a function f that maps arbitrary code words w into the encoding of an associated Amazons position with the following properties:

1. If w encodes a Hex position, then the first player (Black) in decode(w) has a Hex winning strategy if and only if the first player in decode($f(w)$) (Black) has an Amazons winning strategy.

2. Otherwise, $f(w)$ encodes a fixed Amazons position with a second player (White) winning strategy.

3. There exists a Turing machine F and a polynomial p such that for all w, F started on input w computes $f(w)$ in at most $p(|w|)$ steps.

It follows that $w \in$ HEX $\Leftrightarrow f(w) \in$ AMAZONS and HEX $\leq_p$ AMAZONS. We now describe the transformation f.

From a given $n \times n$ Hex position, each of the n^2 hex tiles are represented by a rectangular room containing one Black

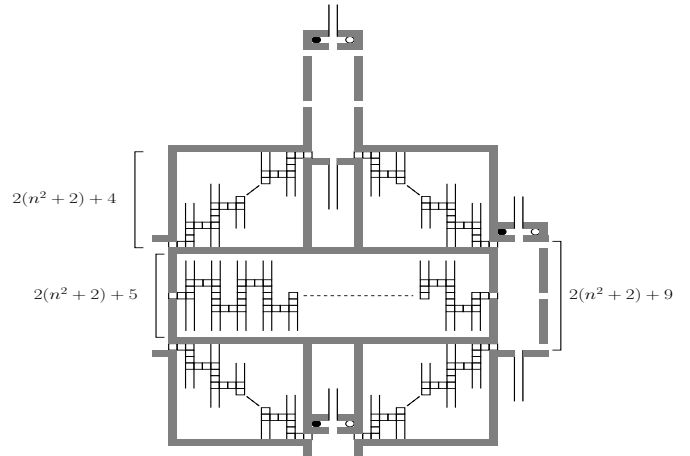[1]We drop "generalized" from now on when the context is clear.

Figure 2: The arrangement of rooms relative to each other and the connecting corridors. There is slack space for arrows in each corridor and at the top and bottom of each room sufficient to allow a large number of amazons to pass through.

and one White amazon (Fig. 2,3). These amazons are placed across from each other so that each is able to contain the other in one move, and so gain control of the room. Occupied tiles on the Hex board are represented by removing the opposing piece. In this way the first stage of the transformed game is effectively a game of Hex where players alternately select rooms, blocking the opponent's amazons. Indeed, if any player does not take control of a room when such a move is available then the opposing player may take a room "for free," which in Hex (and similarly here) can only help.

The rooms are connected by corridors which require a certain minimum number of steps to traverse. This traversal time for each corridor is chosen such that no adjacent rooms may affect either player from securing a room. Each of the n^2 rooms may be sealed off from enemy rooms by blocking off the appropriate corridors. As there are at most 6 entrances to each room and each can be blocked in one move, minimum corridor traversal times of $7n^2$ are sufficient.

To force the outcome of the Amazons game to be equivalent to that of the initial Hex game, White must win if and only if he can form a connecting path from top to bottom. This is accomplished by placing an army of $n^2 + 2$ White amazons at the bottom of the board with a path to all of the bottom rooms (Fig. 4). As with adjacent rooms, these amazons are also sufficiently far away that they cannot influence the initial choosing of rooms.

At the top of the board are $2n^2 + 5$ strips of open space. The size of these strips is chosen large enough so that each is larger than the rest of the board when ignoring the other strips. This ensures that the outcome of the game is entirely determined by which player is able to control a majority of the strips. Strips are positioned such that once a White amazon has captured one, it is unable to capture any others. At the top of each strip is a long path connecting to a single Black amazon (Fig. 5). The path is made long enough so that while the amazon may eventually reach the strip, the number of moves required to do so is greater than that needed by White to cap-
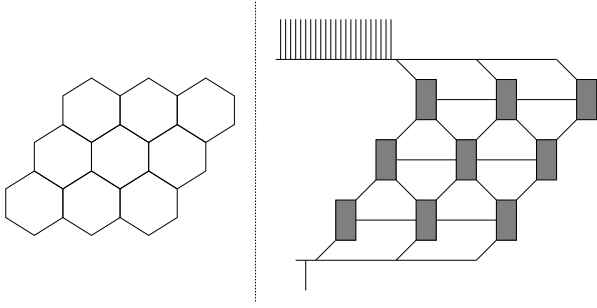
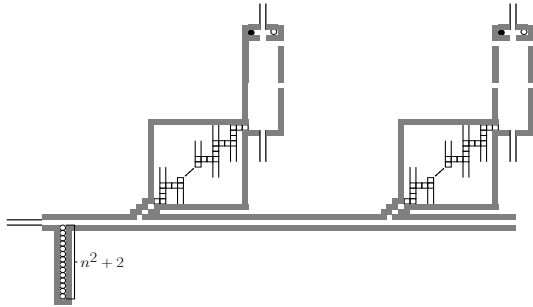Figure 3: Sample transformation of a 3×3 Hex board.

Figure 4: White's army at the bottom of the board. The first amazon can block off paths to enemy rooms.
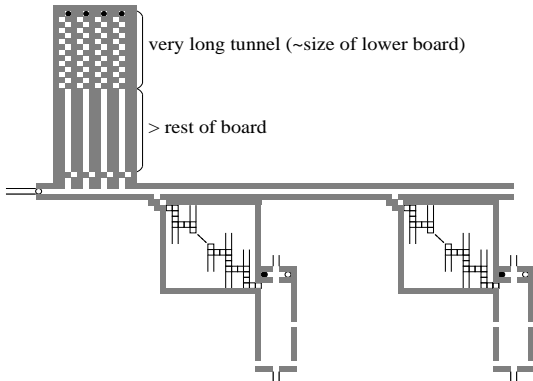

Figure 5: $2n^2 + 5$ large strips of space at the top of the board. White has a gatekeeper to prevent enemy rooms from reaching them. Black has an amazon at the top of each strip that will eventually be able to take control of it if White cannot.

ture as many strips as he can. This is the time needed to decide the control of each room, secure those rooms, and to move the army from the bottom of the board to each of the strips. A very loose upper bound for this distance is the number of free squares in the lower section of the board, everything beneath the strips. In this manner Black is eventually able to take control of any strips that White cannot.

If the first player (White) has a winning strategy in the initial Hex position that player may capture a set of rooms such that a path exists from the bottom of the playfield to the top. By blocking all corridors to hostile rooms he may isolate this path from enemy influence. Since there are no draws in Hex, a winning strategy for the second player is equivalent to preventing White from forming such a sequence of rooms.

If White can form a connecting path then the one White amazon at the top and the army (of size $n^2 + 2$) at the bottom may be moved one at a time to take control of $n^2 + 3$ strips, a majority allowing White to win the game.

If White cannot form a connecting path then the number of strips he may capture is at most $n^2 + 1$: the number of White amazons on the board not counting the army. Black will then eventually be able to take control of the remaining strips with the amazon at the top of each. This gives Black a majority of free space allowing him to win. The winner of this game is therefore the same as for the original Hex position.

The connecting corridors must allow for the entire army to pass through them. The given structure requires long slack tunnels so that arrows may be fired while moving. These tunnels must be at least as long as the size of the army, $n^2 + 2$. Not counting the boundary walls, the height of the horizontal corridors then is $2(n^2 + 2) + 5$, and the diagonal corridors one less. Ensuring that the corridors are long enough to prevent room interference is a simple matter of adding additional bends in the path, with each bend increasing the travel time by 2. Since each corridor only needs to be $7n^2$ moves long constructing this new board is clearly polynomial.

Each room is a rectangle with slack tunnels running through the middle. This tunnel is easily sufficient with almost twice the space of those in the diagonal corridors. Moving an amazon through a room is straightforward: move to the center column when entering, then move to the exit row, and finally into the next corridor. To enter and exit on the same row simply move straight across. The initial amazon in the room may be considered as part of the army or moved out of the way to the side.

Rooms and corridors may clearly be arranged without conflict (Fig. 2). The entire transformation takes polynomial time in the size of the original Hex board. The strategy to move the army to the top of the board when a connecting series of rooms exists is then trivial. It is also clearly impossible to do this when a connecting series of rooms does not exist.   □

3 Geography Reduction

Here we show a stronger result, namely that even if the number of Amazons is bounded by a constant, the problem remains PSPACE-complete. We again start with definitions of the sets we consider:

Definition 2:

- **AMAZONS-2** is a set of succinct encodings of all $n \times n$ Amazons positions with exactly one Black and one White amazon and a winning strategy for Black (to move).

- **GEOGRAPHY-BP3** is a set of succinct encodings of GEOGRAPHY positions — consisting of a directed bipartite planar graph, whose node types are shown in Fig. 7,8,9, and a distinguished start node (Fig. 6) with two leaving edges — for which the first player has a winning strategy.
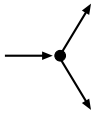
Figure 6: Start vertex.

Figure 7: Path vertex.

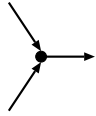

Figure 8: Branch vertex.

Figure 9: Join vertex.

<u>Theorem 2:</u> GEOGRAPHY-BP3 $\leq_p$ AMAZONS-2.

<u>Corollary 2:</u> AMAZONS-2 and AMAZONS are PSPACE-complete.

<u>Proof:</u> Analog to Proof of Corollary 1. Note that AMAZONS-2 $\leq_p$ AMAZONS. □

<u>Proof of Theorem 2:</u> Geography is a game played by two players on a given directed graph with a marked start node. The players alternately put a marker on any unmarked node from which there exists an edge from the node marked last. The first player who cannot move loses. Note that in case of GEOGRAPHY-BP3, the graph is bipartite and planar with node types that are shown above. The game always ends in a vertex with two entering edges and one leaving edge at the second visit of that vertex. [Lichtenstein and Sisper, 1980] show that GEOGRAPHY-BP3 is PSPACE-complete.

We present a polynomial-time transformation that maps input words — which are now encodings of planar bipartite graphs with a start marker — into encodings of Amazons positions with two amazons, such that the first player wins the geography game if and only if the first player wins the associated Amazons game. We emulate the Geography game with two amazons — one for each player — and with blocks that fill the board with the necessary structure. The structure has two separate paths for each edge of the given graph. One path (Fig. 10) is for player one and the other path is for player two. The two amazons simultaneously proceed along the paths as quickly as possible, otherwise one player locks up the other player. We construct the structure such that player one winning in generalized geography is equivalent to the amazon of player one being able to trap the amazon of player two. We embed the given directed graph in the board using a pair of paths to represent an edge. To embed the vertices we use two gadgets that force the players to proceed along the paths in pairs and in the same direction as the original directed edges.

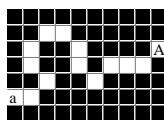Putting a marker on a vertex in GEOGRAPHY-BP3 corresponds to the two amazons going from a vertex gadget to another vertex gadget. Since each vertex gadget can be used only once, vertices cannot be marked twice. We force the two amazons to go along the same paths. If one amazon advances toward a path different from the path chosen by the acting player (the player placing the marker) then the amazon will be locked out by the acting player at some gadget.

We implement a path, which has a one-square width, as blank squares with surrounding blocked squares, as shown in **Fig. 10.** We assume an amazon is currently at the left entrance a and proceeds to the right exit A. At each player's respective turn, the amazon is moved forward and blocks a square behind it. If the player blocks a square in front of him the amazon becomes trapped. The amazons need to travel at maximum speed, otherwise they will be locked out by the opponent's amazon at some point. In Fig. 10, the amazon needs at least 12 moves to travel from the entrance to the exit. The amazon can also get there in 13 moves. We use gadgets that force the amazons to travel in the minimal numbers of moves (in this case 12). Thus, amazons have to traverse any straight line section of a path in one move. If we want to force an amazon to take several moves, we bend the path an appropriate number of times as shown in Fig. 10. Moreover, it is obvious that we can change the direction a path aims as we like. We now discuss the sub-gadgets shown in Fig. 11 to 14.

Type 1 top speed gadget (Fig. 11). Assume the amazon for player X (denoted X) comes to the bottom left entrance, a, and the amazon for player Y (Y) comes to the bottom right entrance, b. This gadget forces Y to come to the entrance as quickly as possible. That is, if Y comes to the entrance just after X comes to the entrance (we say that they arrive at the entrances at the same time in this situation), then X cannot trap Y and X should exit this gadget at A, and Y should exit at B. If Y comes to this gadget late then X can lock Y out in 3 moves by moving to E, C, then D and blocking F. Thus Y should come to the entrance as quickly as possible. If Y comes to entrance b at the same time X enters, Y can move to F and block D. Thus, Y cannot be trapped. In this case both X and Y can exit this gadget in five moves; X's exit is A, Y's exit is B. Note that if we use two paths to connect this gadget to a symmetrical one, we can force both X and Y to move at top speed along those paths.

Type 2 top speed gadget (Fig. 12). If X enters this gadget from a and Y enters from b (resp. d and c), the gadget forces X to proceed to exit A and Y to proceed to exit B (resp. D and C). Otherwise the player being late will be trapped by the other player. This gadget is somewhat complex, so first we will simply refer to its functionality. Assume amazon X comes to the entrance a and amazon Y comes to the entrance b (resp. d and c). This gadget forces Y to come to the entrance
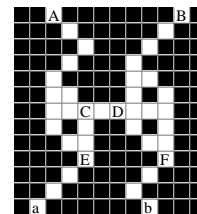

Figure 10: Path example.

Figure 11: Top speed gadget (type 1).

as quickly as possible. If Y's entrance is b, this is the same as type 1 top speed gadget, and Y should come to b at the same time that X enters. If Y's entrance is d, this gadget forces Y to come to d at least before X reaches G, or otherwise Y will be locked out by X. If Y's entrance is c, then even if Y comes to c at the same time that X enters, X can trap Y. In addition, if X enters from b (resp. c) and Y enters from c (resp. b), Y should come to c at the same time that X enters. Otherwise X can lock Y out. If X's entrance is a and Y's entrance is b, X is forced to exit at A and Y is forced to exit at B. Amazons can use the gadget twice, once to enter at a and b and once to enter at d and c. At the first use of the gadget they cannot put blocks to obstruct the next use of the gadget.

Now we detail the amazons' sequence of moves in this gadget. First we will describe the moves when X's entrance is a and Y's entrance is b and they come to the entrance at the same time. X moves to N, Y moves to Q; X then moves to G and blocks I, Y moves to O and blocks H. Then they proceed to their respective exits at top speed. If X blocks a position other than I, Y can then proceed to other exits. This increases the number of moves available to Y, so X should not play this way if he wants to win. Thus we can assume that X blocks I. If Y blocks a position other than H, X can trap Y. Thus Y should block H to prevent being locked out. If X's entrance is a and Y's entrance is c and they arrive at the entrance at the same time, X moves to N and Y moves to R. X then moves to G and blocks L, Y moves to P and either blocks F (option 1) or blocks any other position (option 2). In the case of (1), X moves to K and blocks M. In the case of (2), X moves to J and blocks E. In both cases, Y is locked
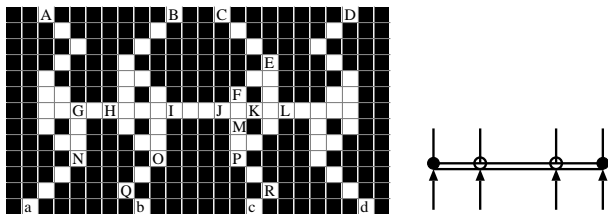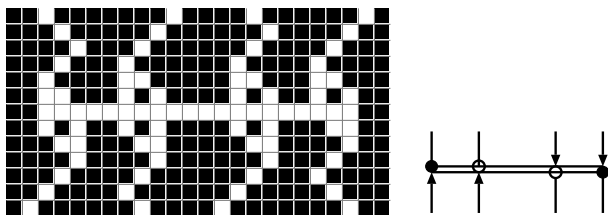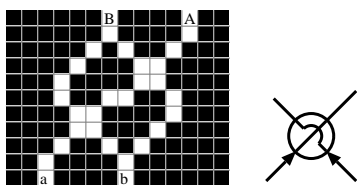
out by X. We can check the other properties analogously.

Type 3 top speed gadget (Fig. 13). This gadget is almost identical to the previous gadget. The only difference is the direction of the right hand side paths which have been mirrored.

Cross gadget (Fig. 14). We can cross a path of player X and a path of player Y with this gadget. Assume the entrance for X is a, and the entrance for Y is b. If X and Y come to the gadget at the same time, X is forced to exit the gadget at A and Y is forced to exit this gadget at B. Both X and Y take five moves in that case. If X comes later, then X will be locked out. If Y comes two moves later than X, then Y will be trapped. Similarly, if a player goes to another exit, or tries to obstruct the opponent player, the opponent player can lock the amazon out. In addition, if X comes to the gadget and Y never comes, then if X exits at B, X takes two more moves than if X exits at A. If Y comes to the gadget and X never comes, then if Y exits at A, Y takes two more moves than if Y exit at B. We can easily check these properties.

We now explain the main vertex gadgets corresponding to the four types of vertices in Fig. 6, Fig. 7, Fig. 8 and Fig. 9. It is easy to emulate a vertex in Fig. 7. We do this with two paths, one for each player, on which player one needs one more move to travel from entrance to exit than player two does, effectively reversing which player is next to act. We illustrate gadgets for branching vertices (Fig. 8) and joining vertices (Fig. 9) in Fig. 15 and 16, respectively. We can use the gadget in Fig. 15 for the start vertex (Fig. 6), too. We draw paths for amazon X, who is expected to enter this gadget first, as solid lines and paths for amazon Y as dotted lines. Two amazons always come to the paired entrances at the same time, otherwise an amazon being late will be locked out by the opponent's amazon at the top speed gadget just after the entrance.

Branch gadget (Fig. 15). This gadget corresponds to a vertex with an entering edge and two leaving edges (Fig. 8), from which a player selects one of the leaving edges to follow. Since we can adjust the number of moves amazons require along paths by bending these paths arbitrarily, we set those from X's entrances to X's exits and those from Y's entrances to Y's exits to be the same length. Because of the cross gadget, X requires two more moves if X wants to take Y's exit, therefore doing so will result in X being trapped by Y in the type 2 top speed gadget. Thus player X cannot select Y's path. Y must select the exit in the same direction that X did, otherwise Y is locked out at the type 2 top speed gadget. Note that if X slows down inside the gadget Y can lock X

Figure 12: Top speed gadget (type 2).

Figure 13: Top speed gadget (type 3).
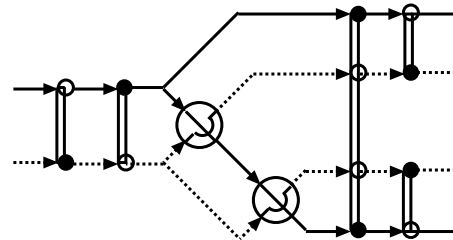

Figure 14: Cross gadget.
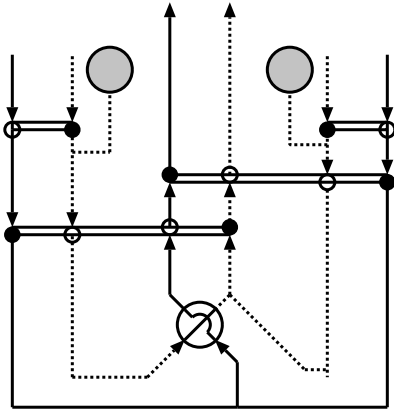

Figure 15: Branch gadget.

Figure 16: Join gadget.

out in the cross gadget or any of the top speed gadgets. In addition, we can use this gadget for the start vertex (Fig. 6). We place Black and White amazons at the entrance so that the Black amazon would enter the gadget first. The size of this gadget is clearly independent of the input size.

Join gadget (Fig. 16). This gadget corresponds to a vertex with two entering edges and a leaving edge (Fig. 9). Amazons come to this gadget at the left or right entrance. If it is the amazons' first visit to this gadget, the gadget forces the amazons to exit at the top center. If one player changes his way and advances to the entrance not used, the opponent's amazon can trap him in a type 2 top speed gadget in front of the entrance. At the second visit, the amazons cannot exit the gadget, because they can use the left exit of the last top speed gadget only once. Thus, after they pass the first top speed gadget, they have to play Amazons only in the gadget. In this case, Y will win by entering the region denoted by the gray ball, which has larger space than that of the rest of this gadget, and closing the entrance to the ball. Y only has to wait till X fills up the rest of the gadget and becomes locked out by himself. This situation corresponds to that in GEOGRAPHY-BP3 where it is player X's turn and the adjacent vertices are already marked. The size of this gadget is also independent of the input size.

Using the gadgets described above we can transform any bipartite planar graph — as described in the definition of GEOGRAPHY-BP3 — into an Amazons position such that the player to move wins the Geography game if and only if the first player wins the Amazons game. First we embed the given graph into a grid, taking space polynomially bounded by the input graph's size using a linear time orthogonal graph drawing algorithm (e.g. [Kant, 1996]). Then we replace each grid point which corresponds to a vertex or an edge with a vertex gadget or a path gadget respectively. Each vertex is incident to leaving edges and entering edges at some directions: north, south, west, and east. We can set the entrances and exits of the vertex gadgets equal to the directions of these edges by surrounding them with path gadgets. Further, we sometimes have to use the cross gadget to connect exits to the right entrances in the next gadget. Vertex gadgets and the path gadgets surrounding them are drawn in a board of constant size. An edge corresponding to a grid point can be also drawn in a constant size, even if it includes a cross gadget. Thus, each grid point is drawn in a board of a constant size, and the size of the Amazons game board is bounded by a polynomial in the size of the input planar graph. To complete the transformation function definition with respect to the three properties stated in the previous proof, input words that do not encode valid graphs are mapped into a fixed Amazons position with two opposing amazons, Black to move and White to win. From the above explanations, it is clear that the run-time of the entire transformation is polynomial in the input length. □

4   Conclusion

We have presented two reductions which imply that determining the winner of generalized Amazons positions is PSPACE-complete. The Hex reduction is appealing because it directly relates two board game complexities. The Geography reduction, however, leads to a stronger result, because the transformation only generates Amazons positions with two opposing amazons, whereas in the Hex reduction, the number of generated amazons cannot be bounded above by a constant.

As this paper was being finished we discovered that Robert Hearn had simultaneously derived similar results [Hearn, 2005]. His reduction shows that generalized Amazons with an unbounded number of amazons is PSPACE-complete.

References

[Berlekamp, 2000] E. Berlekamp. Sums of $n \times 2$ amazons. In *Institute of Mathematics, Statistics Lecture Notes – Monograph Series 35*, pages 1–34. Springer, 2000.

[Buro, 2000] M. Buro. Simple amazons endgames and their connection to hamilton circuits in cubic subgrid graphs. In *Computers and Games Conference*, pages 250–261. Springer, 2000.

[Hearn, 2005] R.A. Hearn. Amazons is PSPACE–complete. *http://arxiv.org/abs/cs.CC/0502013*, 2005.

[Kant, 1996] G. Kant. Drawing planar graphs using the canonical ordering. *Algorithmica*, 16(1):4–32, 1996.

[Lichtenstein and Sisper, 1980] D. Lichtenstein and M. Sisper. Go is polynomial–space hard. *J. of the ACM*, 27:393–401, 1980.

[Lieberum, 2005] J. Lieberum. An evaluation function for the game of amazons. *Theoretical Computer Science*, accepted, 2005.

[Müller and Tegos, 2002] M. Müller and T. Tegos. Experiments in computer amazons. In R. Nowakowski, editor, *More Games of No Chance*, pages 243–260. Cambridge University Press, 2002.

[Reisch, 1981] S. Reisch. Hex ist PSPACE–vollständig. *Acta Informatica*, 15:167–191, 1981.

[Snatzke, 2004] R.G. Snatzke. *Exhaustive search of databases in the application of Combinatorial Game Theory to the game [of] amazons*. Ph.D. thesis, University of Augsburg, Germany, 2004.