One-color Triangle Avoidance Games*

Steven C. Cater
Department of Science and Mathematics
Kettering University
Flint, MI 48504-4898

Frank Harary
Department of Computer Science
New Mexico State University
Las Cruces, NM 88003-8001

Robert W. Robinson
Department of Computer Science
The University of Georgia
Athens, GA 30602-7404

Abstract

The results of exhaustive analysis of the game digraphs for triangle avoidance on $n \leq 12$ nodes are reported. The outcome for $n = 12$ is the first counterexample to a conjecture that would have provided a solution for all n. Computational methods and related games are also discussed.

*This is a preprint of an article accepted for publication in *Congressus Numerantium*

1 Introduction

A *triangle* is a 3-cycle of a graph. The *one-color triangle achievement game* on n nodes starts with n isolated nodes. There are two players A and B; the first move is always made by A who chooses any two of the nodes and joins them by an edge. Then B draws another edge using the same color, after which A draws a third edge of the same color, and so forth. The player who first completes a triangle wins the game. Since all edges have the same color, the triangle can be formed from edges contributed by both players, as in [4].

We call the player who has a winning strategy the *Winner*. Then for one-color triangle achievement, as noted by Seress [6], we have

$$A \text{ is the Winner for } n \equiv 2 \text{ or } 3 \ (mod\, 4), \text{ and}$$

$$B \text{ is the Winner for } n \equiv 0 \text{ or } 1 \ (mod\, 4).$$

Obviously triangle achievement has the same Winner as the game of avoiding two adjacent edges, *i.e.*, the path P_3. Hence the players draw independent edges as long as possible, giving the outcome above.

We concentrate henceforth on the triangle avoidance game, which is highly nontrivial. In *one-color triangle avoidance*, the players and moves are the same. The only difference is that now the first player who completes a triangle loses the game!

Any avoidance game can be expressed as another, entirely equivalent, achievement game. In this case, the goal becomes to achieve a *maximal triangle-free graph*, that is, a triangle-free graph with the property that the addition of any new edge creates a triangle. In general, for given n it is an unsolved problem as to whether it is A or B who has the winning strategy for the n-node game.

In Section 2 we describe the game acyclic digraph (alas, sometimes called a 'DAG') for triangle avoidance, and in Section 3 we report on computational results which extend our knowledge of the Winners to $n = 12$. (The Winners for $n \leq 9$ were reported by Seress [6].) Related games are discussed in the following section, and some of the computational considerations are described in the final section.

An introduction to the terminology and basic results of combinatorial game theory is given in [1]. For graph theory we refer to [3], with the proviso that the terminology is modernized to *node* in place of vertex or point, *edge* for an undirected line, and *arc* for a directed line. After t moves in one-color triangle avoidance, the state of the game will be some triangle-free graph G on n nodes with t edges. A legal move will add another edge to G to obtain a triangle-free graph $G^{'}$, which is called a *child* of G.

We illustrate these concepts by showing in Figure 1 the game digraph for $n = 4$ nodes. At the top of Figure 1, the graph with no edges is the start of the game. Up to isomorphism, there is only one first move for player A; call this $A1$. Now for move $B1$ there are two possibilities as shown. Thus the graph with just one edge has two children, shown in Figure 1 by small arrows. As can be seen from the position P_4, a child can have more than one parent. The two graphs in Figure 1 having no children are both complete bipartite, and hence maximal triangle-free.

Starting with $n = 5$, there is a maximal triangle-free graph which is not bipartite, namely the 5-cycle, C_5. Further, for all $n \geq 5$, there exists a maximal triangle-free graph of order n that is not bipartite. As n grows, the proportion of maximal triangle free graphs on n nodes which are not bipartite seems to tend to 1. This phenomenon offers an intuitive explanation of the apparent difficulty of determining the Winner of triangle avoidance for an arbitrary number n of nodes.

It should be noted that what we have called the game digraph has traditionally been presented as a game tree by providing separate labeled copies of a game position for each distinct sequence of moves which reaches it. Since we are concerned with computational efficiency, we provide no duplicate positions. Thus the game positions and the single moves from one to another form a digraph. For one-color triangle avoidance this game digraph is acyclic, as detailed at the start of Section 3.

For further efficiency we identify any two positions which are isomorphic since the Winner is obviously preserved by graph isomorphism. The general graph isomorphisim problem is widely believed to be computationally intractable, but for small graphs it has been solved very efficiently by B. D. McKay's software package `nauty` [5]. Isomorphism classes of graphs are also known as *unlabeled graphs*. Henceforth, by the *game digraph* (for some order n) we mean the digraph in which the nodes are unlabeled triangle-free graphs of order n and the arcs join pairs related by adding one new edge and are oriented toward the larger graph.

2 Local and global winners

Following Fraenkel's notation, we partition the set S of all triangle-free graphs into sets N and P, N standing for next and P for previous, as follows. When the graph G has the property that the next player to move has a winning strategy (no matter what moves the other player may make), we put graph G into set N. Otherwise, we put G into set P, since then the previous player has a winning strategy.

As mentioned in the introduction, either A or B is the winner for each n; we call them the *global winners*. On the other hand, at each game position (for example, those in Figure 1), either the next player, N, or the previous player, P, is the Winner from that position; they are called *local winners*. It is computationally convenient to determine the local winner at each game position. Then the global Winner is A if, and only if, the local Winner at the initial position is N. Equivalently, the global Winner is B if, and only if, the local Winner at the initial position is P.

In Figure 1, the two childless game positions C_4 and $K_{1,3}$ are labeled P since the next player must lose by creating a triangle. The parents of these are labeled N, since the next player can choose to move to a P position. Now $2K_2$ is labeled P since its only child is an N position. The parent of $2K_2$ is then labeled N, and the starting position is P. Thus player B is the Winner of triangle avoidance on 4 nodes, as A is next to play at the start.

For arbitrary n we can follow the same procedure as above for $n = 4$, starting with the maximal triangle-free graphs of order n, which are in P. Working our way up from children to parents, we classify each game position on n nodes as lying in P or N, finishing with the edgeless starting position.

For each triangle-free graph G, let $Ch(G)$ denote the set of all children of G. Then we have the general classification rule

$$G \in P \text{ iff } Ch(G) \subseteq N,$$

or equivalently,

$$G \in N \text{ iff } Ch(G) \cap P \neq \emptyset.$$

Figure 1 illustrates for four nodes the game digraph (which is *not* a tree) for triangle avoidance. The nodes of the game digraph are the game positions, with an arc from parent to child for each possible move.

We now describe briefly the winning strategy for B, the global Winner for $n = 4$. This prunes Figure 1 down to the essentials. Let $A1$ denote the first move by A, $B1$ the first move by B, etc. The move $A1$ shown in Figure 2 is the edge 12 joining nodes 1 and 2, which gives the same game position (up to isomorphism) as any other possible first move. Then $B1$, the edge 34, leaves A only one possible reply, again up to isomorphism of the game position. The game ends when $B2$ completes a quadrilateral, leaving A with no move to make which avoids creating a triangle. Hence A resigns.

3 How the winners were found

For each $n \geq 3$ the game digraph is a finite acyclic digraph. This is because every move adds one edge to the game position, and there are only $n(n-1)/2$ possible edges. Recall that Turán's Theorem [3, p. 17] shows that a triangle-free graph on n nodes has at most $n^2/4$ edges, giving an upper bound on the number of moves in any game of triangle avoidance.

It is clear that since the property of not containing a triangle is preserved by graph isomorphism, we can work with the game digraph in which each node is an unlabeled graph. For $n = 4$, Figure 1 shows the game digraph. The seven unlabeled graphs correspond to a total of 41 different labeled graphs on the fixed node set $\{1,2,3,4\}$. For larger n, the ratio of labeled to unlabeled graphs tends to $n!$, so considerable computational saving is offered by the game digraph.

In order to compute the Winner for n nodes using the game digraph, one needs to calculate graph isomorphisms. For n of modest size, Brendan McKay's `nauty` software package [5] effectively solves this problem. The phrase "**no aut**omorphisms, **y**es?" provides the acronymic name `nauty`, referring to the likelihood that no nontrivial automorphism will exist in a random graph. One of the utilities in the package is `makeg`, which stands for "**make g**raphs". We have used `makeg` to generate the unlabeled triangle-free graphs on n nodes for $n \leq 12$. For each n we took the output from `makeg` to build a representation of the game digraph for triangle avoidance on n nodes, then applied the recursive method to determine membership in N or P for all of the game positions back to the starting point. The Winners for $n \leq 12$ computed in this way are presented in Table 1. Some of the computational details of the representation employed are discussed in Section 5.

n	3	4	5	6	7	8	9	10	11	12
Winner	B	B	B	A	B	B	B	A	B	A

Table 1. The Winners of triangle avoidance for $n \leq 12$.

For several years the authors had computed only the data for $n \leq 11$, and so were tempted to conjecture that the Winner for triangle avoidance on n nodes is A iff $n \equiv 2\ (mod\ 4)$. Fortunately, a faster workstation with more memory enabled us to extend the computation to $n = 12$, for which that conjecture fails.

However we are still left with the obvious question:

Query: Is it true that for all but some finite exceptional values of n, the Winner for triangle avoidance is A iff $n \equiv 2\ (mod\ 4)$?

The results of Wanless [7, 8] encourage the conjecture that there will be some periodic pattern with a finite number of exceptions. In the first of these papers Wanless confirmed our conjecture to the effect that the Winner of 4-cycle achievement is determined by the residue class of n modulo 7 for all $n \geq 4$ except $n = 8, 13$ and 25. He built on that in the second paper to show that the game of achieving a path of length 6 has period 84.

4 Related games

The main result of [6] is the solution of the "connected version" of triangle avoidance. In this variation, every edge played after the first move must be adjacent to some previously played edge. It turns out that A is the Winner of connected triangle avoidance if, and only if, n is even. A maximum/minimum version of triangle avoidance is explored in [2].

Another natural related game is that of *odd cycle avoidance*, in which the first player to complete a cycle of odd length loses the game.

Theorem 1. *The Winner of odd cycle avoidance on $n \geq 2$ nodes is A if and only if $n \equiv 2\ (mod\ 4)$.*

Proof. It is well known that a graph G is free of odd cycles if and only if G is bipartite. Thus we may assume that the game is played until the position graph is maximal bipartite, *i.e.*, a complete bipartite graph $K(a, b)$ with $a, b \geq 1$ and $a + b = n$. Since $K(a, b)$ has exactly ab edges, B wins the game if a or b is even, and A wins if a and b are both odd. If n is odd, one of a, b must be even so B is the Winner.

If n is even, say $n = 2m$, then either player can ensure that the maximal bipartite game position attained is isomorphic to $K(m, m)$, so that A is the Winner, if and only if, $n \equiv 2 \ (mod\ 4)$. The strategy for Winner is to manage play so that each connected component of the game position is *balanced* in the sense that the two partite classes are equal in cardinality. Of course in a connected bipartite graph the partition of the node set into two partite classes is unique.

It can be seen inductively that Winner can play so that after each of his moves the balance condition is satisfied. Loser can only create an imbalanced nontrivial component by joining an isolated node to a balanced component B. Since the total number of nodes is even, there is another isolated node remaining, which Winner joins by an edge to some vertex in the larger partite class of B in order to reestablish balance. If Loser presents a balanced game position, Winner preserves balance by joining two isolated nodes to form a new nontrivial balanced component, joining two balanced components to form one larger balanced component, or else adding an edge within an incomplete balanced component. To see that Winner can always do this, note that all nontrivial components are of even order, so that the number of isolated nodes is even. Then Winner would have no balance preserving move only in case the game position graph consists of a single complete balanced bipartite component, which must be isomorphic to $K(m, m)$. But this is impossible as Winner is precisely the player who makes the last of the m^2 moves to complete $K(m, m)$. □

5 Computational aspects

The basic idea for finding winners in the order n triangle avoidance game is simple: first construct the game digraph for order n. Then, using a sweep from the bottom up, label the nodes of the game digraph so as to indicate who wins from that position. (This is the algorithm given in Section 2.) However a naive implementation, using recursive functions in a high-level language, tends to run slowly.

Since the nodes of this game digraph consist only of triangle-free graphs, and since the game is over whenever a triangle must be formed, each game digraph sink node must be edge maximal. This consideration led to the following procedure for finding the Winner of the game for order n:

1. Generate a list of the canonical forms of all of the triangle-free graphs of order n. The canonical forms must have the property that lexicographic order on canonical forms is non-decreasing with respect to the number of edges.

2. Sort the canonical forms by lexicographic order. Associate with each canonical form a character, initially U, which we call its *mark*. (U stands for *unmarked*.)

3. For each canonical form, from last to first in lexicographic order, do (a), (b), or (c) according to its mark.:

   (a) If the mark is U, change it to P. We call this *marking* the graph P. Generate each parent, locate its canonical form, and mark it N. The current graph is maximal triangle-free.

   (b) If the mark is N, generate each parent and if marked U mark it P.

   (c) If the mark is P, generate each parent and mark it N.

4. The Winner for order n is A or B depending on whether the first graph (with 0 edges, the starting position) is marked N or P, respectively.

This procedure was implemented as follows:

1. The file is generated by using `makeg`, from the `nauty` package. One option to `makeg` creates only triangle-free graphs. The default is to use `nauty`'s internal definition of canonical form. We supplied a special output format for `nauty` to write our graphs in one text line, in row major order, prefixed with edge count.

2. The file was sorted using the Unix `sort` utility.

3. For the marking, we read the sorted list into an array of the proper size, and had immediate access to the last graph on the list. Since the entire array was in memory, we could perform parent lookup via a binary search, which was done using the C library `bsearch()` function call. Edge deletion was performed in row major order, and each parent graph was transformed to canonical form using `nauty`.

The running time of our implementation was dominated by the sorting routine. Both time and space posed barriers to extending the computations to order $n = 13$. The numbers t_n of unlabeled triangle-free graphs of order n as reported in [5] are listed in Table 2 for $n \leq 15$. The time and the space requirements for our computational procedure for determining Winner for order n both grow at least as fast as t_n.

n	t_n	m_n	w_n
2	2	1	1
3	3	1	1
4	7	2	1
5	14	3	2
6	38	4	3
7	107	6	4
8	410	10	4
9	1897	16	9
10	12172	31	21
11	105071	61	32
12	1262180	147	91
13	20797002		
14	467871369		
15	14232552452		

Table 2. Numbers of triangle-free graphs

The *size* of a graph is the number of edges in it. One way to view the one-color triangle avoidance game is that A's objective is to reach a maximal triangle-free graph of odd size, whereas B's goal is to reach one of even size. We call these sets of graphs the *natural objectives* of A and B. As noted earlier in describing our computational procedure, maximal triangle-free graphs are exactly those which still have a mark value of U when reached in the main loop. In this way we have counted maximal triangle-free graphs in the course of determining the Winners for $n \leq 12$. The number m_n of order n is listed for those values of n in Table 2, along with the number w_n which are Winner's objective.

We attempted to gain insight into Winner's strategy by modifying the game so as to limit Winner's objectives to some proper subset of her natural objectives. For order 9, for instance, we identified a minimal subset of 5 of B's natural objective graphs which enabled B to win. That is, B has a strategy for reaching one of the five graphs, but not for reaching any fixed subset of four of those graphs. While our experiments along these lines have not yet yielded any general insights, this is still a promising direction for future work toward understanding one-color triangle avoidance.

References

1. A. S. Fraenkel, Scenic trails ascending from sea-level Nim to alpine chess, *Games of no chance* (R. J. Nowakowski, ed.), Math. Sci. Res. Inst. Publ., 29, Cambridge Univ. Press, Cambridge (1996), 13-42.

2. Z. Füredi, D. Reimer, and Á. Seress, Hajnal's triangle-free game and extremal graph problems, *Congr. Numer.* **82** (1991), 123-128.

3. F. Harary, *Graph Theory*, Addison-Wesley, Reading, MA (1969).

4. F. Harary, Achievement and avoidance games for graphs, *Ann. Discrete Math.* **13** (1982), 111-119.

5. B. D. McKay, *nauty* user's guide (Ver. 1.5), technical report TR-CS-90-02, Computer Science Dept., Australian National University, 1990. `http://cs.anu.edu.au/people/bdm/`

6. Á. Seress, On Hajnal's triangle-free game, *Graphs Combin.* **8** (1992), 75-79.

7. I. M. Wanless, On the periodicity of graph games, *Australas. J. Combin.* **16** (1997), 113-123.

8. I. M. Wanless, Path achievement games, *Australas. J. Combin.*, to appear.

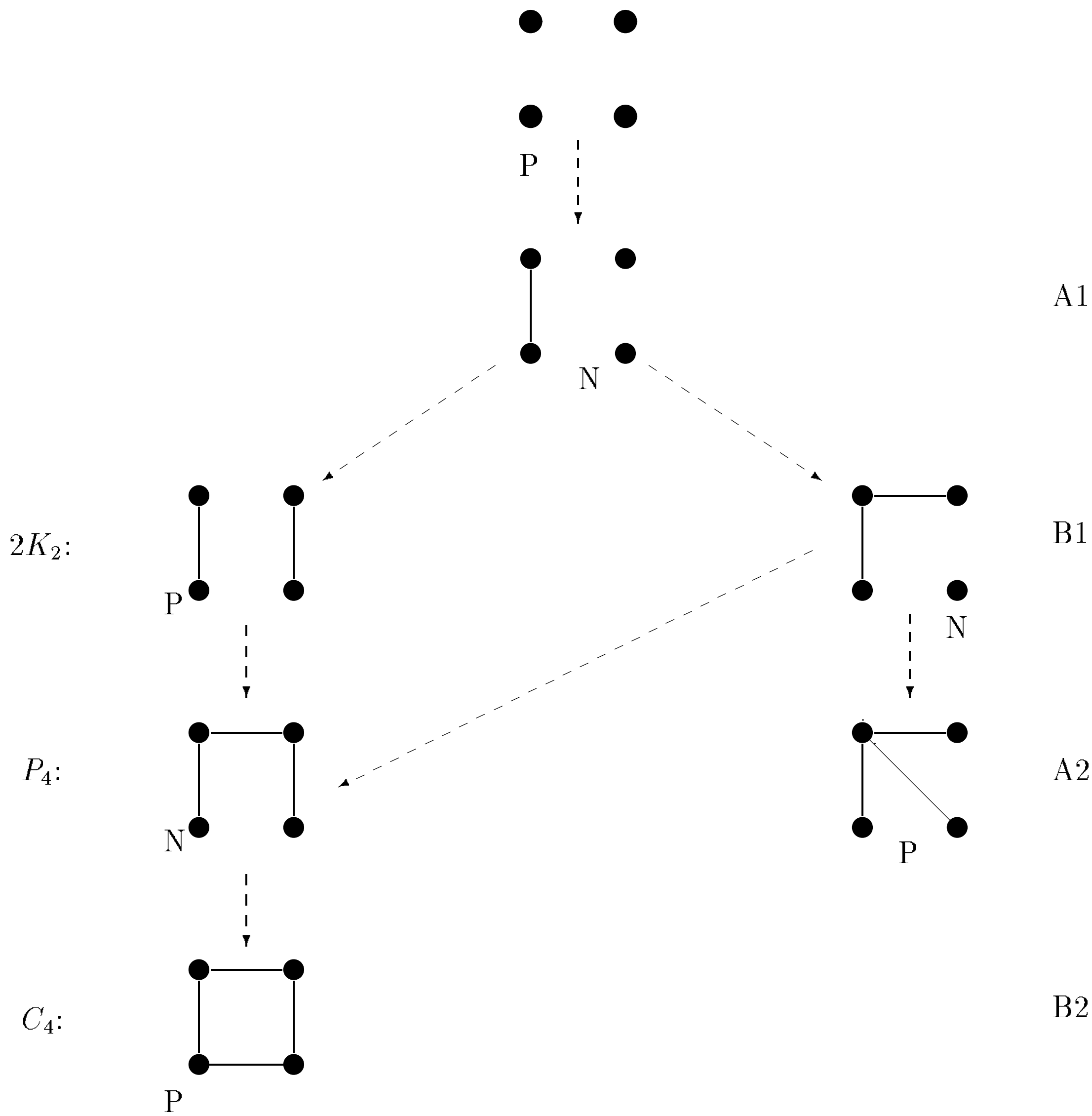

Figure 1: The game digraph for triangle avoidance on four nodes. Here, B wins by choosing the first of the two possible B1 moves.

move	A1	B1	A2	B2	A3
edge	12	34	14	23	resigns

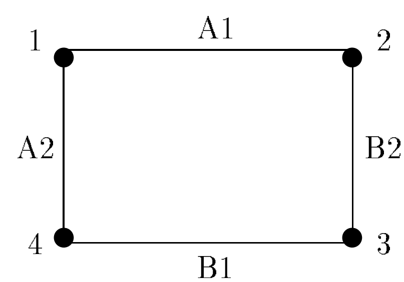

Figure 2: The winning strategy for $n = 4$