

# CycleXings

Andreas Bauer\*, Michael Kaufmann†  
Ignaz Rutter\*, and Julian Vordermeier\*

## 1 Game Idea

CycleXings is a two-player graph-drawing game, where players take turns at positioning the vertices of an  $n$ -vertex cycle to create a straight-line drawing of the cycle. The value  $n$  can be chosen arbitrarily, the default is  $n = 10$ . Each vertex may be positioned exactly once, afterwards its position is fixed. Player 1, the *maximizer* in the first round, seeks to maximize the number of crossings in the drawing. In contrast, Player 2 is the *minimizer*, who seeks to minimize the number of crossings. The first round ends when all vertices are positioned, i.e., after  $n$  moves. In the second round the roles are exchanged. The player creating more crossings when he is playing the maximizer wins the game.

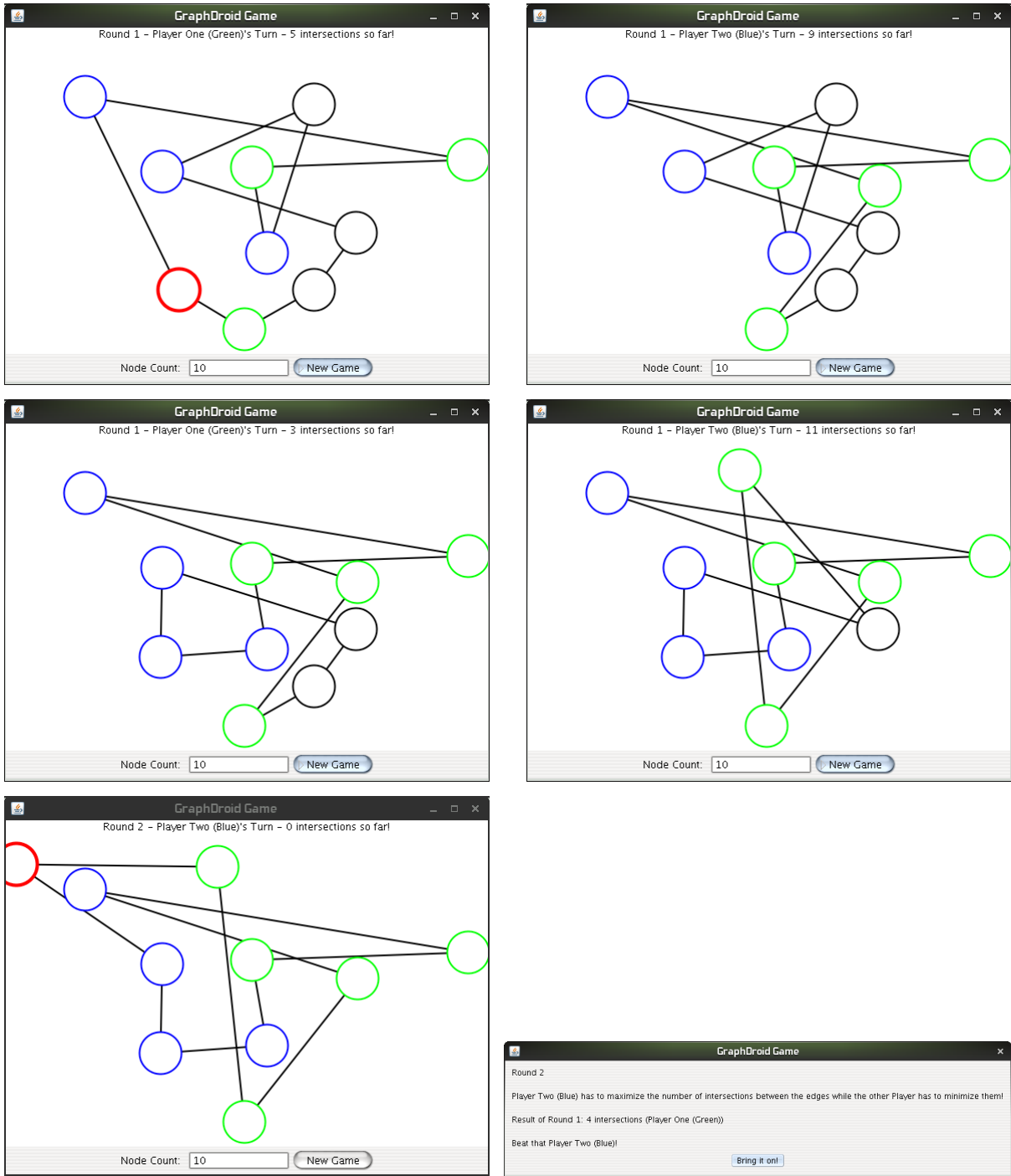
CycleXings can be played on a regular PC running a recent version of Java. However, the better way to experience CycleXings is by using a tablet computer. In addition to the Java variant, we have an Android-based implementation, which can be played on smart phones and tablet computers that support the Android 4 OS.

The following pictures give an impression of the first round of a game. The first picture shows the game state after several moves. Green vertices have been placed by the maximizer, blue nodes by the minimizer. Black nodes have not yet been positioned. The red node is currently selected, it can either be moved and deselected to place it, or another node can be selected for movement. In the latter case the original position of the previously selected node is restored. In the second picture the selected node has been moved up and to the right, and it is now the minimizer's turn. The current number of crossings – 9 in this case – is displayed at the top. In the next picture the minimizer has moved the vertex at the top to the lower left, and has hence decreased the crossings to 3. Again, by moving one vertex to the very top, the maximizer increases to 11 crossings. In the final move of the round, the minimizer moves the last unpositioned node to the upper left, reaching a final crossing number of 4, which is acknowledged with a message box. Then the second round starts, where the second Player (blue) has to maximize the crossings.

---

\*Karlsruhe Institute of Technology (KIT), {andreas.bauer2, julian.vordermeier}@student.kit.edu, ignaz.rutter@kit.edu

†Universität Tübingen, mk@informatik.uni-tuebingen.de



## 2 Relation to Graph Drawing

*Crossing minimization* (in straight-line drawings) is certainly one of the most fundamental problems in graph drawing and has been shaping the research area of graph drawing over the past decades from the early NP-hardness proof [2] to a linear-time FPT algorithms [4] and constant-factor approximations for bounded-degree graphs embeddable in a fixed orientable surface [3]. See the recent survey [1] for a complete treatment. The problem “dual” to crossing minimization, *crossing maximization*, is somehow a natural counterpart and has also been researched. The corresponding graph parameter is called the *obfuscation complexity* [5]. Although, admittedly, crossing maximization lacks the

strong practical motivation of its “primal” counterpart crossing minimization. CycleXings combines these two problems into a single game, creating an additional twist by letting the players follow opposite optimization directions.

It seems an interesting problem to devise optimal strategies for the players, possibly also for more general graphs than  $n$ -vertex cycles.

### 3 Obtaining, Installing, and Running CycleXings

The game can be obtained from our web page <http://i11www.iti.uni-karlsruhe.de/projects/cyclexings/index>. The apk file can be installed on any Android device. It may be necessary to allow installation of third-party files in the settings. For the java version, just download the file `CycleXings.jar` and run it with `java -jar CycleXings.jar`.

### References

- [1] *Handbook of Graph Drawing and Visualization*, chapter Crossings and Planarization. CRC Press, 2012. preprint available from <http://www.cs.brown.edu/~rt/gdhandbook/>.
- [2] M. R. Garey and D. S. Johnson. Crossing number is np-complete. *SIAM Journal on Algebraic Discrete Methods*, 4(3):312–316, 1983.
- [3] P. Hliněný and M. Chimani. Approximating the crossing number of graphs embeddable in any orientable surface. In *Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'10)*, pages 918–927. SIAM, 2010.
- [4] K. Kawarabayashi and B. Reed. Computing crossing number in linear time. In *Proceedings of the 39th Annual ACM Symposium on Theory of Computing (STOC'07)*, pages 382–390. ACM New York, 2007.
- [5] O. Verbitsky. On the obfuscation complexity of planar graphs. *Theoretical Computer Science*, 396(1–3):294–300, 2008.