CATCH THE VIRUS (Develop efficient algorithms : as a pen-and-paper game)

Rules of the game

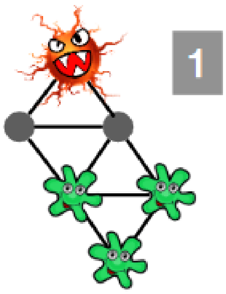
The X13 virus has infiltrated the computer network of the organ donation database. The computer scientist Ada wants to catch X13 by placing several bots on individual nodes of the network.

How many bots does Ada need to catch X13? *(Please keep in mind that the lower the number the bots needed, the better.)*

 The Virus = one red face, virus-like

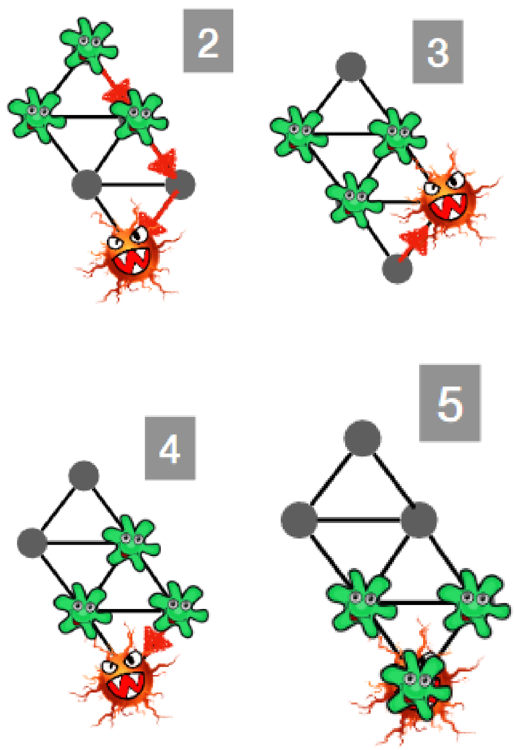
 The bots = green smiling faces

STEP 1



The bots are set on nodes.
The virus sets itself on a free node.

STEP 2



The bots change their position: some of the bots are collected and distributed to new nodes, while the others stay put;

The virus sees what the bots are up to and moves as far as it wants along the network to a new node.

The virus must not cross a node where a bot has been left behind.

Bots that have moved can be skipped.

Networks to be used

Please download the networks at: https://www.ada.wien/wp-content/uploads/2019/09/Graphs_Virus_bot_VCLA_TUWIEN.pdf

EXPLANATION OF THE GAME AND ITS RELEVANCE FOR COMPUTER SCIENCE

This pen-and-paper game called “Catch the virus’’ is a **tree decomposition game** based on the Cops and Robbers Game (Seymour and Thomas, 1993), which was specially adapted by Professor Stefan Szeider, Vienna University of Technology (TU Wien). The game is disseminated by the Vienna Center for Logic and Algorithms and the project ADA – Algorithms Think Differently (www.ada.wien).

Aim of the game

This game is characterised by **Pursuit-Evasion method**, which refers to a series of problems in mathematics and computer science in which one group attempts to track down members of another group in an environment. Tactical problems include avoiding enemies, reaching enemies and catching and sending / reacting to misleading signals. Torrence Parsons introduced the formulation of the **environment modelled as a graph**. Networks where a few bots are sufficient have a certain structure that also allows to solve heavy optimization problems efficiently.

Therefore, **computing the smallest number of bots that is sufficient for a given network** is of great importance for the development of efficient algorithms.

Optimization

Optimization problems are everywhere, and especially so in engineering of algorithms: Balancing design trade-offs is an optimization problem, as are scheduling and logistical planning. The theory — and sometimes the implementation — of control systems relies heavily on optimization, and so does machine learning, which has been the basis of most recent advances in artificial intelligence.

In mathematics, computer science and economics, an **optimization problem** is the **problem** of finding the best solution from all feasible solutions. The original classic real-life example is to lay out a shop floor with multiple workstations and workers, so as to minimize travel distance of the work piece given a required sequence of operations. COMAP's Mathematical Modeling Handbook - as sources for optimization problems encountered (either literally, such as when you want to divide up a cake, or mathematically, when you see another problem to which you can apply these models/methods).

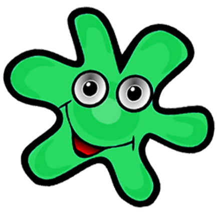
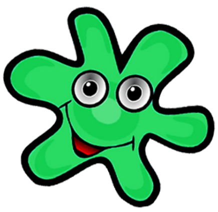
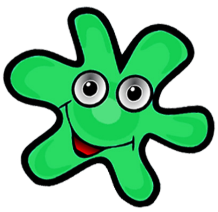
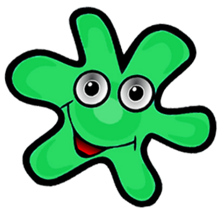
CUT-OUTS



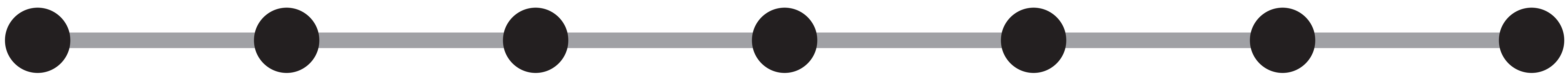


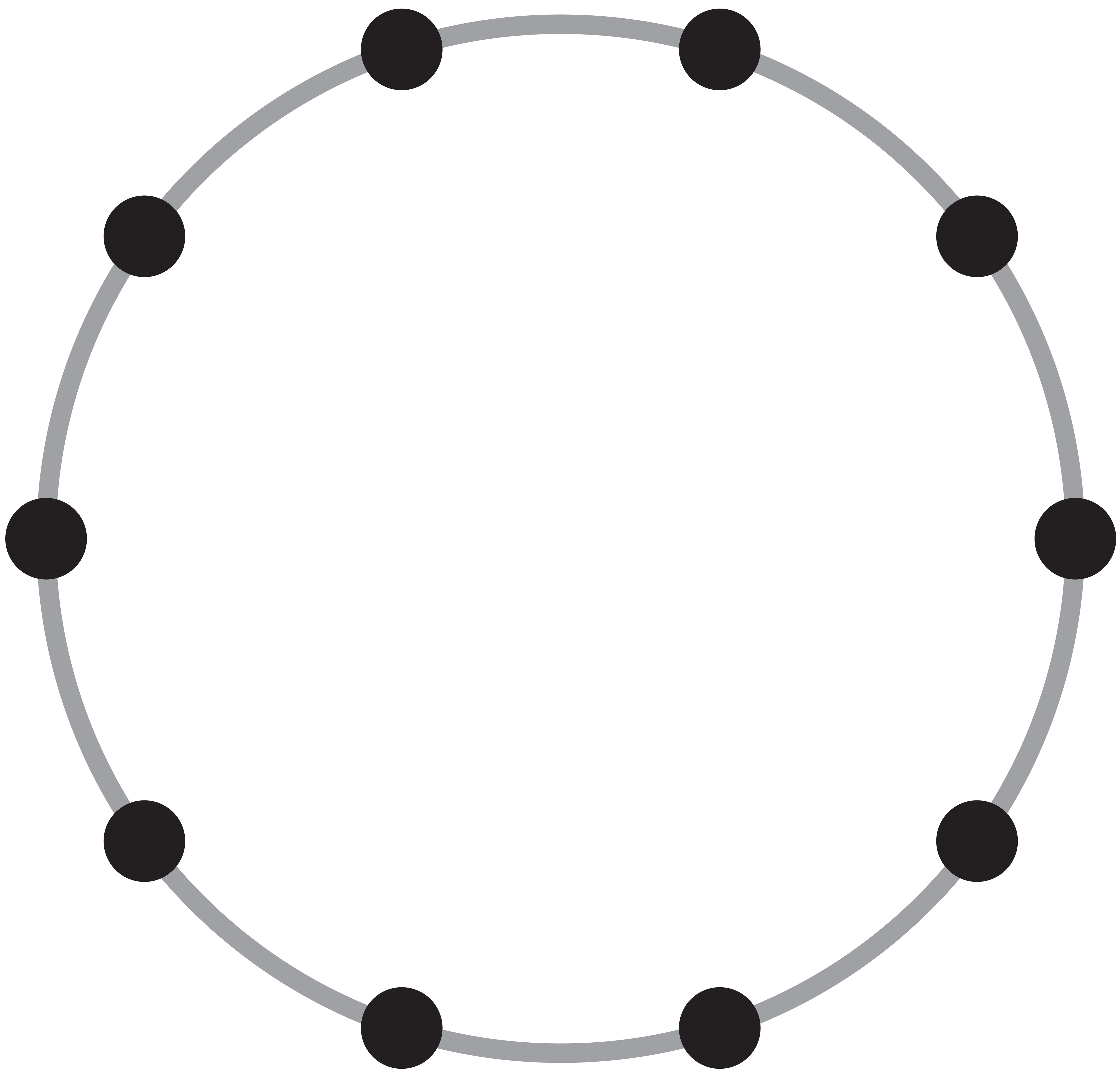










TU WIEN
FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics
Vienna Center for Logic and Algorithms