Publications of Stefan Neubert

This document lists all peer-reviewed publications of Stefan Neubert, Chair for Algorithm Engineering, Hasso Plattner Institute, Potsdam, Germany. This listing was automatically generated on November 29, 2022. An up-to-date version is available online at hpi.de/friedrich/docs/publist/neubert.pdf.

Conference papers

[1] Friedrich, T., Ihde, S., Keßler, C., Lenzner, P., Neubert, S., Schumann, D., Efficient Best Response Computation for Strategic Network Formation under Attack. In: *Symposium on Algorithmic Game Theory (SAGT)* (2017), pp. 199–211, 2017.

Inspired by real world examples, e.g. the Internet, researchers have introduced an abundance of strategic games to study natural phenomena in networks. Unfortunately, almost all of these games have the conceptual drawback of being computationally intractable, i.e. computing a best response strategy or checking if an equilibrium is reached is NP-hard. Thus, a main challenge in the field is to find tractable realistic network formation models. We address this challenge by investigating a very recently introduced model by Goyal et al. [WINE'16] which focuses on robust networks in the presence of a strong adversary who attacks (and kills) nodes in the network and lets this attack spread virus-like to neighboring nodes and their neighbors. Our main result is to establish that this natural model is one of the few exceptions which are both realistic and computationally tractable. In particular, we answer an open question of Goyal et al. by providing an efficient algorithm for computing a best response strategy, which implies that deciding whether the game has reached a Nash equilibrium can be done efficiently as well. Our algorithm essentially solves the problem of computing a minimal connection to a network which maximizes the reachability while hedging against severe attacks on the network infrastructure and may thus be of independent interest.

[2] Friedrich, T., Ihde, S., Keßler, C., Lenzner, P., Neubert, S., Schumann, D., Brief Announcement: Efficient Best Response Computation for Strategic Network Formation under Attack. In: *Symposium on Parallelism in Algorithms and Architectures (SPAA)*, pp. 321–323, 2017.

Inspired by real world examples, e.g. the Internet, researchers have introduced an abundance of strategic games to study natural phenomena in networks. Unfortunately, almost all of these games have the conceptual drawback of being computationally intractable, i.e. computing a best response strategy or checking if an equilibrium is reached is NP-hard. Thus, a main challenge in the field is to find tractable realistic network formation models. We address this challenge by establishing that the recently introduced model by Goyal et al.[WINE'16], which focuses on robust networks in the presence of a strong adversary, is a rare exception which is both realistic and computationally tractable. In particular, we sketch an efficient algorithm for computing a best response strategy, which implies that deciding whether the game has reached a Nash equilibrium can be done efficiently as well. Our algorithm essentially solves the problem of computing a minimal connection to a network which maximizes the reachability while hedging against severe attacks on the network infrastructure.

[3] Teibrich, A., Mueller, S., Guimbretière, F., Kovacs, R., Neubert, S., Baudisch, P., Patching Physical Objects. In: *User Interface Software and Technology (UIST)*, pp. 83–91, 2015.

Personal fabrication is currently a one-way process: Once an object has been fabricated with a 3D printer, it cannot be changed anymore; any change requires printing a new version from scratch. The problem is that this approach ignores the nature of design iteration, i.e. that in subsequent iterations large parts of an object stay the same and only small parts change. This makes fabricating from scratch feel unnecessary and wasteful. In this paper, we propose a different approach: instead of re-printing the entire object from scratch, we suggest patching the existing object to reflect the next design iteration. We built a system on top of a 3D printer that accomplishes this: Users mount the existing object into the 3D printer, then load both the original and the modified 3D model into our software, which in turn calculates how to patch the object. After identifying which parts to remove and what to add, our system locates the existing object in the printer using the system's built-in 3D scanner. After calibrating the orientation, a mill first removes the outdated geometry, then a print head prints the new geometry in place. Since only a fraction of the entire object is refabricated, our approach reduces material consumption and plastic waste (for our example objects by 82% and 93% respectively).