Publications of George Skretas

This document lists all peer-reviewed publications of George Skretas, Chair for Algorithm Engineering, Hasso Plattner Institute, Potsdam, Germany. This listing was automatically generated on October 6, 2024. An up-to-date version is available online at hpi.de/friedrich/docs/publist/skretas.pdf.

Journal articles

[1] Michail, O., Skretas, G., Spirakis, P. G., Distributed computation and reconfiguration in actively dynamic networks. In: *Distributed Computing* 35, pp. 185–206, 2021.

[2] Michail, O., Skretas, G., Spirakis, G. P., On the transformation capability of feasible mechanisms for programmable matter. In: *Computer and System Sciences*, pp. 18–39, 2019.

In this work, we study theoretical models of programmable matter systems. The systems under consideration consist of spherical modules, kept together by magnetic forces and able to perform two minimal mechanical operations (or movements): rotate around a neighbor and slide over a line. In terms of modeling, there are n nodes arranged in a 2-dimensional grid and forming some initial shape. The goal is for the initial shape A to transform to some target shape B by a sequence of movements. Most of the paper focuses on transformability questions, meaning whether it is in principle feasible to transform a given shape to another. We first consider the case in which only rotation is available to the nodes. Our main result is that deciding whether two given shapes A and B can be transformed to each other is in P. We then insist on rotation only and impose the restriction that the nodes must maintain global connectivity throughout the transformation. We prove that the corresponding transformability question is in PSPACE and study the problem of determining the minimum seeds that can make feasible otherwise infeasible transformations. Next we allow both rotations and slidings and prove universality: any two connected shapes A,B of the same number of nodes, can be transformed to each other without breaking connectivity. The worst-case number of movements of the generic strategy is $\Theta(n^2)$. We improve this to $\mathcal{O}(n)$ parallel time, by a pipelining strategy, and prove optimality of both by matching lower bounds. We next turn our attention to distributed transformations. The nodes are now distributed processes able to perform communicate-compute-move rounds. We provide distributed algorithms for a general type of transformation.

Conference papers

[3] Angrick, S., Bals, B., Friedrich, T., Gawendowicz, H., Hastrich, N., Klodt, N., Lenzner, P., Schmidt, J., Skretas, G., Wells, A., How to Reduce Temporal Cliques to Find Sparse Spanners. In: *European Symposium on Algorithms (ESA)*, 2024.

[4] Deligkas, A., Eiben, E., Goldsmith, T.-L., Skretas, G., Being an Influencer is Hard: The Complexity of Influence Maximization in Temporal Graphs with a Fixed Source. In: *Autonomous Agents and Multi-Agent Systems (AAMAS)*, pp. 2222–2230, 2023.

We consider the influence maximization problem over a temporal graph, where there is a single fixed source. We deviate from the standard model of influence maximization, where the goal is to choose the set of most influential vertices. Instead, in our model we are given a fixed vertex, or source, and the goal is to find the best time steps to transmit so that the influence of this vertex is maximized. We frame this problem as a spreading process that follows a variant of the susceptible-infected-susceptible (SIS) model and we focus on three objective functions.In the MaxSpread objective, the goal is to maximize the total number of vertices that get infected at least once. In the MaxViral objective, the goal is to maximize the number of vertices that are infected at the same time step. Finally, in MaxViralTstep, the goal is to maximize the number of vertices that are infected at a given time step. We perform a thorough complexity theoretic analysis for these three objectives over three different scenarios: (1) the unconstrained setting where the source can transmit whenever it wants; (2) the window-constrained setting where the source has to transmit at either a predetermined, or a shifting window; (3) the periodic setting where the temporal graph has a small period. We prove that all of these problems, with the exception of MaxSpread for periodic graphs, are intractable even for very simple underlying graphs.

[5] Deligkas, A., Eiben, E., Skretas, G., Minimizing Reachability Times on Temporal Graphs via Shifting Labels. In: *International Joint Conference on Artificial Intelligence (IJCAI)* (International Joint Conference on Artificial Intelligence (IJCAI)), pp. 5333–5340, 2023.

We study how we can accelerate the spreading of information in temporal graphs via delaying operations; a problem that captures real-world applications varying from information flows to distribution schedules. In a temporal graph there is a set of fixed vertices and the available connections between them change over time in a predefined manner. We observe that, in some cases, the delay of some connections can in fact decrease the time required to reach from some vertex (source) to another vertex (target). We study how we can minimize the maximum time a set of source vertices needs to reach every other vertex of the graph when we are allowed to delay some of the connections of the graph. For one source, we prove that the problem is W[2]-hard and NP-hard, when parameterized by the number of allowed delays. On the other hand, we derive a polynomial-time algorithm for one source and unbounded number of delays. This is the best we can hope for; we show that the problem becomes NP-hard when there are two sources and the number of delays is not bounded. We complement our negative result by providing an FPT algorithm parameterized by the treewidth of the graph plus the lifetime of the optimal solution. Finally, we provide polynomial-time algorithms for several classes of graphs.

[6] Bilò, D., Cohen, S., Friedrich, T., Gawendowicz, H., Klodt, N., Lenzner, P., Skretas, G., Temporal Network Creation Games. In: *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2511–2519, 2023.

Most networks are not static objects, but instead they change over time. This observation has sparked rigorous research on temporal graphs within the last years. In temporal graphs, we have a fixed set of nodes and the connections between them are only available at certain time steps. This gives rise to a plethora of algorithmic problems on such graphs, most prominently the problem of finding temporal spanners, i.e., the computation of subgraphs that guarantee all pairs reachability via temporal paths. To the best of our knowledge, only centralized approaches for the solution of this problem are known. However, many real-world networks are not shaped by a central designer but instead they emerge and evolve by the interaction of many strategic agents. This observation is the driving force of the recent intensive research on game-theoretic network formation models. In this work we bring together these two recent research directions: temporal graphs and game-theoretic network formation. As a first step into this new realm, we focus on a simplified setting where a complete temporal host graph is given and the agents, corresponding to its nodes, selfishly create incident edges to ensure that they can reach all other nodes via temporal paths in the created network. This yields temporal spanners as equilibria of our game. We prove results on the convergence to and the existence of equilibrium networks, on the complexity of finding best agent strategies, and on the quality of the equilibria. By taking these first important steps, we uncover challenging open problems that call for an in-depth exploration of the creation of temporal graphs by strategic agents.

[7] Mertzios, G. B., Michail, O., Skretas, G., Spirakis, P. G., Theofilatos, M., The Complexity of Growing a Graph. In: *International Symposium on Algorithms and Experiments for Wireless Sensor Networks*, pp. 123–137, 2022.

[8] Michail, O., Skretas, G., Spirakis, G. P., Distributed Computation and Reconfiguration in Actively Dynamic Networks. In: *Principles of Distributed Computing (PODC)*, pp. 448–457, 2020.

In this paper, motivated by recent advances in the algorithmic theory of dynamic networks, we study systems of distributed entities that can actively modify their communication network. This gives rise to distributed algorithms that apart from communication can also exploit network reconfiguration in order to carry out a given task. At the same time, the distributed task itself may now require a global reconfiguration from a given initial network G_s to a target network G_f from a family of networks having some good properties, like small diameter. With reasonably powerful computational entities, there is a straightforward algorithm that transforms any G_s into a spanning clique in $\mathcal{O}(\log n)$ time, where time is measured in synchronous rounds and n is the number of entities. From the clique, the algorithm can then compute any global function on inputs and reconfigure to any desired target network in one additional round. We argue that such a strategy, while time-optimal, is impractical for real applications. In real dynamic networks there is typically a cost associated with creating and maintaining connections. To formally capture such costs, we define three reasonable edge-complexity measures: the total edge activations, the maximum activated edges per round, and the maximum activated degree of a node. The clique formation strategy highlighted above, maximizes all of them. We aim at improved algorithms that will achieve (poly)$log(n)$ time while minimizing the edge-complexity for the general task of transforming any G_s into a G_f of diameter (poly)log(n). There is a natural trade-off between time and edge complexity. Our main lower bound shows that $\Omega(n)$ total edge activations and $\Omega(n/\log n)$ activations per round must be paid by any algorithm (even centralized) that achieves an optimum of $\Theta(\log n)$ rounds. On the positive side, we give three distributed algorithms for our general task. The first runs in $\mathcal{O}(logn)$ time, with at most $2n$ active edges per round, an optimal total of $\mathcal{O}(n \log n)$ edge activations, a maximum degree $n-1$, and a target network of diameter 2. The second achieves bounded degree by paying an additional logarithmic factor in time and in total edge activations, that is, $\mathcal{O}(\log^2 n)$ and $\mathcal{O}(n \log^2 n)$, respectively. It gives a target network of diameter $\mathcal{O}(\log n)$ and uses $\mathcal{O}(n)$ active edges per round. Our third algorithm shows that if we slightly increase the maximum degree to polylog(n) then we can achieve a running time of $\wr(\log^2 n)$. This novel model of distributed computation and reconfiguration in actively dynamic networks and the proposed measures of the edge complexity of distributed algorithms may open new avenues for research in the algorithmic theory of dynamic networks. At the same time, they may serve as an abstraction of more constrained active-reconfiguration systems, such as reconfigurable robotics which come with geometric constraints, and draw interesting connections with alternative network reconfiguration models, like overlay network construction and network constructors. We discuss several open problems and promising future research directions.

[9] Michail, O., Skretas, G., Spirakis, G. P., On the Transformation Capability of Feasible Mechanisms for Programmable Matter. In: *International Colloquium on Automata, Languages and Programming (ICALP)*, pp. 136:1–136:15, 2017.

In this work, we study theoretical models of programmable matter systems. The systems under consideration consist of spherical modules, kept together by magnetic forces and able to perform two minimal mechanical operations (or movements): rotate around a neighbor and slide over a line. In terms of modeling, there are n nodes arranged in a 2-dimensional grid and forming some initial shape. The goal is for the initial shape A to transform to some target shape B by a sequence of movements. Most of the paper focuses on transformability questions, meaning whether it is in principle feasible to transform a given shape to another. We first consider the case in which only rotation is available to the nodes. Our main result is that deciding whether two given shapes A and B can be transformed to each other is in P. We then insist on rotation only and impose the restriction that the nodes must maintain global connectivity throughout the transformation. We prove that the corresponding transformability question is in PSPACE and study the problem of determining the minimum seeds that can make feasible otherwise infeasible transformations. Next we allow both rotations and slidings and prove universality: any two connected shapes A,B of the same number of nodes, can be transformed to each other without breaking connectivity. The worst-case number of movements of the generic strategy is $\Theta(n^2)$. We improve this to $\mathcal{O}(n)$ parallel time, by a pipelining strategy, and prove optimality of both by matching lower bounds. We next turn our attention to distributed transformations. The nodes are now distributed processes able to perform communicate-compute-move rounds. We provide distributed algorithms for a general type of transformation.