どんな研究？

Classical max-flow \( \rightarrow O(|Edges| \times |Nodes|) \) algorithm

Network Interdiction
1. Attacker strikes
2. Max-flow is computed
3. NP-hard (for any \( k \) attacks)

Adaptive Network Flow
1. Flow is computed
2. Attacker strikes
3. NP-hard (for any \( k \) attacks)

Voronoi Diagram
Input: set \( P \) of points in \( \mathbb{R}^2 \)
Output: Subdivision of \( \mathbb{R}^2 \), such that each region has a common nearest neighbor in \( P \).
Output format: vertices of Voronoi diagram in arbitrary order

Takes \( O(n \log n) \) time using \( O(n) \) space \( O(n^2) \) time using \( O(1) \) space [Asano et al. 2011]
Result: \( O(n^{3/2} \log n + n \log n \log \log n) \times \) time in \( O(s) \) space

何がわかる？

時間と空間のトレードオフ
・最大流問題
・3人寄れば文殊の知恵？
・ネットワークの時系列解析
・条件付き経路探索
・行列のランク計算

研究内容

Limited Memory

Started in the 70’s

Model

Input: read-only, random-access
Memory: \( O(k) \) words
Output: write-only

Increased interest recently

Andre’ van Renssen

Jean-Francois Baffier

Multiroute Flow as a tool

Motivation: disjoint routes with the same value
Introduces in 1993 by Wataru Kishimoto et al.
Fast to compute and we improve it!
\((k+1)-\)Approximation and we prove it!
Quite good in practice and we evaluate it!
Solve Interdiction problems for some graphs
At least \( 97\% \) of success on classic families of graph: grids, complete networks, random-shape networks
Weak on complex networks (R-MAT generator)
Give a much better approximation and improved bounds
Upper bound: deterministic (and \( k \)-approximation to Adaptive Network Flow)
Lower bound: Bilevel Optimization
Our method minimizes the energy/ressources consumption

Voronoi Diagram

Input: set \( P \) of points in \( \mathbb{R}^2 \)
Output: Subdivision of \( \mathbb{R}^2 \), such that each region has a common nearest neighbor in \( P \).
Output format: vertices of Voronoi diagram in arbitrary order

Takes \( O(n \log n) \) time using \( O(n) \) space \( O(n^2) \) time using \( O(1) \) space [Asano et al. 2011]
Result: \( O(n^{3/2} \log n + n \log n \log \log n) \times \) time in \( O(s) \) space