Graph and Distributed Algorithms

Euripides Markou

Department of Computer Science & Biomedical Informatics, University of Central Greece, Lamia, Greece.

Bilateral Collaboration between University of Athens and INRIA Sophia-Antipolis 3rd Workshop, 21-22 May 2012, University of Athens
Outline

1. Research - Collaborations
2. Black-Hole Search in Networks
3. Autonomous Selfish Users in Communication Networks
4. Computational Biology
Research interests

Design of algorithms and study the computational complexity for problems especially in the areas of:

- Distributed Computing
- Algorithmic Game Theory
- Computational Geometry
- Computational Biology
Recent students at the Univ. of Central Greece (Bachelor Theses)

Collaborating groups through the years

- A. Pelc, Université du Québec en Outaouais, Gatineau, Canada. (I was a postdoc, 2003-2004).
- I. Emiris, University of Athens. (I was a postdoc, 2004-2006).
- R. Klasing, LaBRI, Bordeaux, France. (I was a postdoc, 2006).
- G. Karakostas, McMaster University, Hamilton, Canada. (I was a postdoc, 2006-2007).
Collaborating groups through the years

- Comenius University and Slovak Academy of Sciences, Bratislava, Slovakia, February 2009 and October 2011.
- McMaster University, Hamilton, Canada, March 2010 and August 2010.
- Laboratoire d’Informatique Fondamentale de Marseille (LIF), Marseille, France, November 2010.
- École Centrale Marseille and LIF, Marseille, France, November 2011.
PYTHAGORAS (Univ. of Athens, Apr. 2004 - Dec. 2007)

Title: ‘Design and development of geometric algorithms for curved objects’. Funded by the Greek Ministry of Education.
Coordinator: Prof. Ioannis Emiris.
We used Computational Geometry methods to study curved objects in the plane and the 3d space, which are encountered in geometric design, graphics and modeling, in structural biology and bioinformatics, in robot navigation among obstacles, and in network and VLSI design.
Mobile and Swarm Robotics (Univ. of Central Greece, June 2011)

Title: ‘Mobile and Swarm Robotics’. Funded by the Canadian Government and the Univ. of Central Greece. The purpose was to fund a one month visit of Dr. Michel Paquette (Vanier College, Montreal, Canada) at the Univ. of Central Greece. We studied the Black-Hole Search problem in an unoriented torus.
Research - Current Projects

THALES: GeomComp (Univ. of Athens, Apr. 2012 - Sep. 2015)

- Title: ‘Advanced Geometric Computing and Critical Applications’. Funded by the Greek Ministry of Education.
- Coordinator: Prof. Ioannis Emiris.
- Includes three Greek teams from Univ. of Athens, National Technical Univ. of Athens and Univ. of Crete and researchers from abroad.
THALES: GeomComp (Univ. of Athens, Apr. 2012 - Sep. 2015)

We study a modern, multi-disciplinary approach, at the interaction of Computer Science, Engineering, and Computational Mathematics.

- We expect to deliver top-level algorithmic results for representative and important problems in the domain, along with robust implementations, leading to the practical solution of specific, critical applications.
- Our open source software, often integrated in the CGAL library, should be instrumental for dissemination and educational purposes.
THALES: AlgoNOW (National Technical Univ. of Athens, Apr. 2012 - Sep. 2015)

- Coordinator: Prof. Stathis Zachos.
- Includes three greek teams from National Technical Univ. of Athens, Univ. of Athens, and Athens Univ. of Economics and Business and researchers from abroad.
THALES: AlgoNOW (National Technical Univ. of Athens, Apr. 2012 - Sep. 2015)

We investigate fundamental problems that arise from the increasing needs of computation and communication. Research topics:

- Location, availability, and processing of massive data sets: approximation and online algorithms.
Research - A couple of new on-going works

**Periodic Traversals in Graphs**
With J. Chalopin, A. Labourel, LIF, Marseille, France.
Algorithms for periodic traversals of nodes of a graph.

**On-line Graph Exploration with Advice**
With S. Dobrev, R. Kralovic, Slovak Academy of Sciences, and Comenius University, Bratislava, Slovakia.
Algorithms for on-line graph exploration with advice. We improved the best known lower bound on the competitive ratio for deterministic online algorithms, and we gave an on-line algorithm with a constant competitive ratio using a short advice. The first results will be presented at SIROCCO 2012.
Black Hole Search (BHS)

- Network (Connected Graph $G$)
- Mobile agents
- Dangerous node (Black Hole): destroys any agent visiting that node

Black Hole Search
Explore $G$ and locate $BH$

Optimize:
- # moves
- # agents
The agents:

- are **anonymous** (same deterministic algorithm)
- have **constant** memory
- carry a **constant number of tokens**
- see other agents or tokens when they are at the **same** node.
- are **synchronous**
- are initially **scattered** in the network
- have **no knowledge** about the size of the network or the total number and initial positions of the agents
Network Exploration

**Dangerous Network**
- Team of agents
- Co-located, distinct identities
- Synchronous or Asynchronous
- Communication:
  - whiteboards
  - tokens-on-nodes
  - tokens-on-edges

**Safe Network**
- Single agent
- Restricted Memory (logarithmic)
- Constant memory + Tokens
- Undirected / Directed
- How much memory?
- How many tokens?
Question

Can agents having constant memory explore dangerous graphs and locate BH?

Yes if:
- they are co-located and synchronous
- the graph can be explored (e.g., a ring)

Time-out mechanism

next node is safe  next node is the BH
BHS with constant memory agents

Question
Can agents having **constant memory** explore dangerous graphs and locate BH?

Yes if:
- they are co-located and synchronous
- the graph can be explored (e.g., a ring)

Time-out mechanism
- next node is safe
- next node is the BH
**Question**

Can agents having constant memory explore dangerous graphs and locate BH?

**Yes if:**
- they are co-located and synchronous
- the graph can be explored (e.g., a ring)

**Time-out mechanism**

- next node is safe
- next node is the BH
BHS with constant memory agents

Question
Can agents having constant memory explore dangerous graphs and locate BH?

Yes if:
- they are co-located and synchronous
- the graph can be explored (e.g., a ring)

Time-out mechanism

next node is safe
next node is the BH
BHS with constant memory agents

Question
Can agents having \textit{constant memory} explore dangerous graphs and locate BH?

Yes if:
- they are co-located and synchronous
- the graph can be explored (e.g., a ring)

Time-out mechanism

next node is safe

next node is the BH
BHS with constant memory agents

Question
Can agents having constant memory explore dangerous graphs and locate BH?

Yes if:
- they are co-located and synchronous
- the graph can be explored (e.g., a ring)

Time-out mechanism

next node is safe
next node is the BH
**Question**

Can agents having *constant memory* explore dangerous graphs and locate BH?

**Yes if:**

- they are co-located and synchronous
- the graph can be explored (e.g., a ring)

**Time-out mechanism**

- next node is safe
- next node is the BH
BHS with scattered constant memory agents

Question
Can we solve the BHS problem with scattered constant memory agents using tokens?

Yes in:
- Anonymous Rings (cycles $n \geq 3$)
  - oriented: all agents agree on the ring's orientation
  - non-oriented: agents may disagree on the orientation
- Anonymous Torus (toroidal grids $n \times m$, $n, m \geq 3$)
The agents agree on the cardinal directions (N,S,W,E)
Question
Can we solve the BHS problem with scattered constant memory agents using tokens?

Yes in:
- Anonymous Rings (cycles $n \geq 3$)
  - oriented: all agents agree on the ring’s orientation
  - non-oriented: agents may disagree on the orientation
- Anonymous Torus (toroidal grids $n \times m$, $n, m \geq 3$)
The agents agree on the cardinal directions (N,S,W,E)
Minimizing the # of agents and tokens

Our goals

- Find the minimal resources to solve the problem: # of agents and # of tokens per agent
- Find trade-offs between these two resources
- Evaluate the impact of constant memory on the resources needed
- Observe the differences between movable (reusable) tokens and unmovable tokens.
Our results on the ring (*SIROCCO ’11*)

<table>
<thead>
<tr>
<th>Tokens are</th>
<th>Ring is</th>
<th># agents</th>
<th># tokens</th>
</tr>
</thead>
<tbody>
<tr>
<td>Movable</td>
<td>Oriented</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Unoriented</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Unmovable</td>
<td>Oriented</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Unoriented</td>
<td>5</td>
<td>2</td>
</tr>
</tbody>
</table>

- Lower bounds work even if agents have non-constant memory.
- No trade-off between # of agents and # of tokens.
Our results on the torus (*DISC ’11*)

- **Impossibility results**:  
  - No algorithm for any constant number of agents with any constant number of unmovable tokens  
  - No algorithm for 3 agents with 1 movable token each  
  - No algorithm for 2 agents (with any number of tokens), even if they have unlimited memory

- **Algorithms**:  
  - An algorithm for \(\geq 3\) agents with 3 tokens  
  - An algorithm for \(\geq 4\) agents with 2 tokens  
  - An algorithm for exactly 3 agents with 2 tokens
Some recent related results

Problem

Design efficient wireless networks of devices which take upon themselves certain network responsibilities.

Ad-hoc networks vs networks with a central authority.

Selfishness of each node: the effort of the node to maximize its own utility without caring about the results of its actions on the overall network-wide outcome.
Solutions

Offer reward to nodes for their cooperation or punishment for non-cooperation.

**Incentives as reward or punishment**

- **Micro-payment schemes**: distribution of credit to nodes. Usually, the distribution and/or the expenditure of credit is controlled by a central authority [L., Buttyan and J.-P. Hubaux, 2003], [S., Zhong, J., Chen, Y. R. Yang, 2003].

**Our model (I)**

<table>
<thead>
<tr>
<th>Strategy of a node $u$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. the amount and routing of the flow originating at $u$,</td>
</tr>
<tr>
<td>2. the amount of the flow that $u$ forwards (not originating at $u$),</td>
</tr>
<tr>
<td>3. a non-negative <em>threshold</em> value for each outgoing edge</td>
</tr>
</tbody>
</table>
Our model (II)

Abstraction of the reputation mechanism:
Consider an edge \((x, y)\) of the network and suppose that \(x\) forwards to \(y\) a total flow \(\Phi\) with final destinations, nodes different than \(y\). Suppose also that a part \(\epsilon\) of \(\Phi\) is cut by \(y\). If \(\epsilon\) is more than the threshold value that \(x\) keeps for \(y\), then \(x\) disconnects edge \((x, y)\).

Utility of a node
- increases with the flow originating at or destined for this node and reaches its destination,
- decreases with the flow sent out or forwarded by this node.
Our questions

Given a network and a set of demands:

- Can we decide in polynomial time whether there exists a Nash-equilibrium so that a non-zero quantity of every demand is satisfied?
- Can we compute in polynomial time (some) node strategies that, once imposed, will lead the system to a (connected) Nash-equilibrium?

Nash-equilibrium:

No node has a profit to change its strategy while all other nodes maintain their own strategies.
Results

We characterize the complexity of computing initial values that lead to a connected Nash-equilibrium in our protocol:

**Necessary and sufficient conditions**
for the existence of *non-trivial* and *connected* Nash-equilibria.

**Decide the existence of a connected Nash-equilibrium**

- We present a LP algorithm which decides whether a connected Nash-equilibrium *without unsuccessful* flows exists and if so, the algorithm calculates (in polynomial time) the values that impose such an equilibrium on the network.
- Finally, for the general case we prove that it is NP-hard to decide whether a *connected* Nash-equilibrium exists.
Open Questions

Open questions

- Find topologies where a connected Nash-equilibrium can be decided in polynomial time. (Can be done for example in trees.)
- How easy is to decide whether a selection of specific initial values for the nodes’ decision variables will lead the game to a Nash-equilibrium?
- What about the speed of the convergence?
- What is the price of anarchy?

Some of the results

Protein folding prediction

Given a sequence of amino-acids, find a folding which minimizes the total energy.

Hydrophobic-Polar model

Given: A sequence of hydrophobic and hydrophilic amino-acids
Goal: A folding of the sequence on a lattice so that:
- the order of the amino-acids is preserved and
- the adjacent (unit distance) hydrophobic amino-acids on the lattice are maximized

The problem is NP-hard for 3D [Berger B., Leighton T., 1998] or 2D lattice [Crescenzi et al, 1998] and admits constant approx. algorithms for 2D or 3D lattice [Hart W., Istrail S., 1996].
Proteins usually fold into their native structure very fast, despite the fact that the conformation space is huge. How nature figures out the right folding pathway so fast? (Levinthal’s paradox)

[Sali et al, 1994], [Clote 1999]

The folding time seemed to be small if and only if the energy gap between the lowest energy and the second lowest energy of conformations on the lattice was large.

We probably need a better understanding of the nature of dominant forces.
However, even if the optimal (native) fold in a certain lattice model is found it could be quite far from the real fold of the protein. Identifying lattice models which have a potential to produce folds close to real 3D structures is an important question in structural proteomics studied in many papers.
Identifying ‘good’ lattice models

Protein Chain Lattice Fitting

To measure the accuracy of representation of lattice models, the following procedure is commonly used:

- Select a test set of proteins with known 3D structure.
- Find the closest lattice representation of each protein minimizing the overall distance of the lattice representation to the exact structure (e.g., c-RMS).
- Compute the average c-RMS over all proteins in the test set.

The crucial part of the above procedure is the computation of the closest lattice representation (of the chain) of a given protein structure. This problem is known as Protein Chain Lattice Fitting (PCLF) problem.
Protein Chain Lattice Fitting (PCLF)

**Problem**

*Given* a sequence of $n$ points $(p_1, p_2, \ldots p_n)$, where $p_i = (x_i, y_i, z_i) \in \mathbb{R}^3$, find a new sequence of points $(q_1, q_2, \ldots q_n)$ with the following properties: this new sequence designates a self-avoiding walk on a lattice and has **least** distance from $(p_1, p_2, \ldots p_n)$.

**Recent results**

It has been shown [Manuch J, Gaur D., 2008] that the PCLF problem is NP-complete for specific cubic lattices and a number of approximation algorithms (but without bounded ratios) have been proposed for this problem.
Protein Chain Lattice Fitting (PCLF)

Mathematical formalization:
Given: a sequence of $n$ points $(p_1, p_2, \ldots p_n)$, where $p_i = (x_i, y_i, z_i) \in \mathbb{R}^3$
Goal: find a new sequence of points $(q_1, q_2, \ldots q_n)$ with the following properties:

- $|q_{i+1} - q_i| = 1$ for all $i$
- for every $i \neq j$ we have $q_i \neq q_j$
- for each $i$, $q_i$ belongs to the lattice
- $\sum_{i=1}^{n} |p_i - q_i|$ is minimized
Open Problems

- From a theoretical point of view it would be very interesting to further investigate the complexity for the 2D lattice.
- From a practical point of view, the following important questions remain open:
  - What is the complexity of the problem in different types of cubic lattices (e.g., non-equilateral, non-orthogonal, etc)?
  - Can we design approximation algorithms with bounded ratio?
Thank you for your attention

Web: users.ucg.gr/~emarkou
E-mail: emarkou@ucg.gr