Activities during the sabbatical—Spring 2015 by Bala Ravikumar, Computer Science Department

Teaching Related Activities:

I participated in a workshop as part of SIGCSE (known as LittleFe Buildout) in Kansas City, Missouri. In this workshop, each team built a parallel computer that will be used for instructional and research purpose.

The following is a description of the LittleFe pbuild-out activity:

LittleFe is a portable mini-cluster which is small enough to fit in a suitcase, is light enough to easily move between classrooms, and travels as standard checked baggage to conferences and workshops (a waterproof, wheeled shipping case is included, and all together the package weighs less than 50 pounds). The 2015 buildout will featured LittleFe units: this design utilizes quad-core Intel® Celeron™ processors with Intel Gen7 Graphics chipsets, which support OpenCL programming.

LittleFe's primary focus is turnkey classroom demonstrations of, and exercises in, High Performance Computing (HPC), parallel programming, and Computational/Data Enabled Science and Engineering (CDESE). LittleFe supports shared memory, distributed memory, and GPGPU parallelism.

LittleFe's secondary focus is as a production HPC resource for small institutions that are not yet able to afford or support a full scale cluster: in fact, LittleFe can be used as a gateway to and development platform for full scale HPC resources such as XSEDE.

I have created some projects to implement on this parallel computer. I plan to assign these projects for the courses that I will be teaching in the future semesters.

B) I participated in a workshop on Computer Security (1stCReSTFaculty DevelopmentWorkshop) held July 13-14, 2015 at The George Washington University Graduate Center in Arlington, VA.

The following is a description of the program from CReST web site: The workshop to provide support for faculty members with no experience teaching security topics who want to integrate security content into their CS courses. In response to our call for participation, we received 4 times the number of applicants for which we have funding from the National Science Foundation (NSF). The selection of participants was based on alignment with our NSF project goals and responses to the application questions related to experience, courses taught, and likelihood to implement workshop content. We also considered broader impact factors as outlined...
by NSF.

Participants went through several potential course projects that focus on security related issues and discussed how these topics can be incorporated into various CS courses. The participating faculty are collaborating to create various course projects that can be included in the CS curriculum.

**Research Activities:**

I wrote the following papers for publication to various journals and conferences:

1) Approximating a nonregular language by Finite Automata (joint with J. Combs, SSU undergraduate student)

   **ABSTRACT:** Regular languages are an important class of languages that is well understood and has wide range of applications in string matching, bioinformatics, deep pocket inspection for network security etc. However, this class is very limited and many important problems can’t be modeled using regular languages or FA. In [an earlier work EiRe], an approach was proposed to approximately accept nonregular languages using FA, and a quantitative measure was introduced to measure the degree of approximation. In this work, we look at two specific non-regular languages for which we study the trade-off between degree of approximation and the size of the recognizing DFA.

   The paper will be submitted to one of the upcoming conferences such as *Conference on Implementation and Application of Automata* or *Descriptional Complexity of Formal Systems*.

2) An improved Upper-bound for Rivest et al.'s Half-lie Problem

   A preliminary version of this paper appeared in Theory and Applications of Models of Computation (TAMC 2014). The revised version I worked on during the sabbatical has been submitted for publication in ACM Transactions on Algorithms.

   **ABSTRACT:**

   Ulam proposed a problem of determining an optimum strategy for searching for an integer x in \{1, 2, …, n\} using binary queries (i.e., queries with yes/no answer) in which the responses to up to k queries (for fixed k) can be incorrect. This problem has been extensively studied for the past fifty years. The paper by Rivest et al. [9] that made a major advance in Ulam’s problem in 1981 introduced a restricted type of error in responses known as half-lies. Rivest et al.
presented a lower-bound on the minimax complexity of half-lie version of Ulam's search problem. Here we present an analysis of the half-lie problem in the case of \( k = 1 \) and \( k = 2 \) and present an upper-bound that improves the bound of Rivest et al. for all sufficiently large values \( n \). In particular, the original problem of Ulam in which \( n \) is one million, the best known prior upper-bound was 25. The bound presented here improves it to 24.

3) Comparing Physarum Networks and Random Networks (joint with students Z. Dahlgren and K. Vasconcellos)

Physarum is a mold that can form a network of protoplasmic veins, and many nuclei during a phase of its life cycle when it searches for food. The network it constructs seems to have strongest edge weights on the shortest path between food sources. Tero, Kobayashi, Nakagaki, Meholhorn and others have proposed analytical models to explain the way physarum grows. In this work, we use image processing technique to automatically extract a network from the photograph of the petri dish in which we grew physarum and compare the network to random graphs (of Erdos-Renyi type) as well as the network arising from social interactions (such as Strogatz-Watts networks). It is shown that the physarum networks more closely resemble the latter than the former.

4) On Parikh Membership Problems for Finite Automata, Push-down Automata and Counter Machines.

(A preliminary version of this paper appeared in the conference Languages and Automata Theory LATA 2014 and a revised version has been completed for journal submission.)

Abstract

We consider the problem of determining if a string \( w \) belongs to a language \( L \) specified by an automaton (NFA, or PDA augmented by reversal-bounded counters, etc.) where the string \( w \) is specified by its Parikh vector. If the automaton (PDA augmented with reversal-bounded counters) is fixed and the Parikh vector is encoded in unary (binary), the problem is in \textit{DLOGSPACE (PTIME)}. When the automaton is part of the input and the Parikh vector is encoded in binary, we show the following results: if the input is an NFA accepting a letter-bounded language (i.e., \( \subseteq a_1 \ldots a_k^* \) for some distinct symbols \( a_1, \ldots, a_k \)), the problem is in \textit{PTIME}, but if the input is an NFA accepting a word-bounded language (i.e., \( \subseteq w_1 \ldots w_m^* \) for some nonnull strings \( w_1, \ldots, w_m \)), it is \textit{NP}-complete. The proofs involve solving systems of linear Diophantine equations with non-negative integer coefficients. As an application of the results, we present efficient algorithms for a generalization of a tiling problem posed recently by Dana Scott. Finally, we give a classification of the complexity of the membership problem for restricted classes of semilinear sets.

Keywords: Parikh vector, NFA, counter machine, reversal-bounded counters, CFG, Chomsky Normal Form, bounded language