▼ Introduction to Algorithmic Challenges in Emerging Applications of Computing Minitrack

Linda Morales
Computer Science Department,
Texas A&M University-Commerce
Linda_Morales@tamu-commerce.edu

Wolf Bein
School of Computer Science
University of Nevada, Las Vegas
bein@cs.unlv.edu

Algorithmic challenges arise in many emerging areas of computing: *security, bioinformatics,* and *algorithmic game theory* are examples. In the area of algorithmic game theory, agents may be in varying degrees of collaboration and competition, and questions about equilibria arise. In the area of power management, improvements in battery tech-nology for mobile devices may lag behind improvements in hardware. Online algorithms may allow power management schemes to schedule resources without full knowledge of future demands. In information security, secure dissemination of information, cyberforensics, and prevention of cybercrime are in demand.

The minitrack on Algorithmic Challenges in Emerging Applications of Computing explores algorithmic challenges involving online and randomized algorithms, scheduling theory, approximation algorithms, optimization, and algorithmic complexity.

Nine papers will be presented. In *Bounding Prefix Transposition Distance for Strings and Permutations*, Chitturi and Sudborough improve the upper and lower bounds for prefix transposition distance to n-$\log_8$n and 2n/3, respectively. They give upper and lower bounds for the prefix transposition distance on strings, and they prove that the prefix transposition distance problem on strings is NP complete. In *Finding a minimum-sum dipolar spanning tree in R^3,* Daescu and Bitner give an algorithm for finding the minimum-sum dipolar spanning tree in R^3 that takes $O(n^2\log^2 n)$ time using $O(n^2)$ space.

In *How Many Attackers Can Selfish Defenders Catch?* Mavronicolas, Monien and Papadopoulou study distributed systems with attacks and defenses, and address the fundamental question of determining the maximum amount of protection achievable by a number of such defenders against a number of attackers if the system is in a Nash equilibrium. They identify graph-theoretic thresholds for the number of defenders that determine the possibility of optimizing the Defense-Ratio and obtain a comprehensive collection of trade-off results.

In *K-Trunk and Efficient Algorithms for Finding a K-Trunk on a Tree Network,* Li, Peng and Chu develop efficient algorithms for finding the k-trunk of a tree, where given an edge-weighted tree T, a k-trunk is a subtree T_k with k leaves in T, which minimizes the sum of the distances of all vertices in T from T_k plus the weight of T_k.

In their paper *On Finding Bicliques in Bipartite Graphs: a Novel Algorithm with Application to Integration of Diverse Biological Data Types*, Zhang, Chesler, and Langston present a fast and novel algorithm that finds all maximal bicliques in a bipartite graph. In *Towards Fast Incremental Hashing of Large Bit Vectors,* Wilder and Jeffery present an incremental hashing technique that performs well in application domains that require algorithmic support for collections of large bit vectors that are frequently referenced.

In *Clustering and the Biclique Partition Problem*, Bein, Bein, Meng, Morales, Shields and Sudborough show that the Biclique Partition Problem (BPP) does not have a polynomial time α-approximation algorithm, for any $\alpha \geq 1$, unless P=NP, and that BPP, restricted to whether at most k cliques are sufficient, has a polynomial time 2-approximation algorithm. They give a O(VE) time algorithm for BPP(2).

In *Knowledge States: A Tool for Randomized Online Algorithms,* Bein, Larmore, and Reischuk give a new model to derive competitive randomized online algorithms and to analyze the tradeoff between competitiveness and memory. In *Priority Approximation for Batching,* Bein, Noga and Wiegley give the first constant approximation ratio algorithm for serial batch scheduling. They also use adaptive techniques to give lower bounds.