รูปแบบ Abstract (บทคัดย่อ)

Project Code : RSA5480006
(รหัสโครงการ)

Project Title : ความซับซ้อนของการให้ป้ายบนกราฟบางอย่าง (The Complexities of Some Graph Labelings)
(ชื่อโครงการ)

Investigator : รศ.ดร.สรรพวรรธน์ กันตะบุตร (Sanpawat Kantabutra)
(ชื่อนักวิจัย)

E-mail Address : sanpawat@alumni.tufts.edu

Project Period : 15 มิถุนายน 2554 ถึง 14 มิถุนายน 2557
(ระยะเวลาโครงการ)

Abstract

This research project is composed of two parts. In the first part, given an undirected, connected, simple graph $G = (V,E)$, two vertex labelings L_V and L'_V of the vertices of G, and a label *flip* operation that interchanges a pair of labels on adjacent vertices, the VERTEX RELABELING PROBLEM is to transform G from L_V into L'_V using the flip operation. Agnarsson et al. showed solving the VERTEX RELABELING PROBLEM on arbitrary graphs can be done in $\Theta(n^2)$, where n is the number of vertices in G [3]. In this part we study the VERTEX RELABELING PROBLEM on graphs $K_{m,m}$ and introduce the concept of *parity and precise labelings*. We show that, when we consider the parity labeling, the problem on graphs $K_{m,m}$ can be solved quickly in O(log m) time using m processors on an EREW PRAM. Additionally, we also show that the number of processors can be further reduced to $\frac{m}{\log m}$ in this case while the time complexity does not change. When the labeling is precise, the parallel time complexity increases by a factor of *log m* while the processor complexities remain m and $\frac{m}{\log m}$. We also show that, when graphs are restricted to $K_{m,m}$, this problem can be solved optimally in O(m) time when the labeling is parity, and can be solved in O(m log m) time when the labeling is precise, thereby improving the result in [3] for this specific case. Moreover, we generalize the result in the case of precise labeling to the cases when L_V and L'_V can be any configuration. In the second part we consider a form of graph labeling that is applied to a layered-security model in which the containers and their nestings are given in the form of a rooted tree T. As part of the input to the problem, we are given lists of *penetration costs* for the containers and *target-acquisition values* for the prizes contained within the containers. We want to assign the penetration costs to the edges and the target-acquisition values to the vertices of the tree T to minimize the total prize that an attacker can acquire given a fixed *budget*. The attacker breaks into containers starting at the root and once a vertex has been broken into its children can be broken into by paying the associated penetration cost. The attacker must deduct the corresponding penetration cost from the budget as each new container is broken into. We conjecture that in general it is not possible to develop an *optimal defensive strategy*. Therefore, we give an algorithm for producing a *good defensive strategy* ---one meeting certain desirable properties, but that cannot be guaranteed to be optimal.