A Maximum Weighted Flow Algorithm Based on the Push-Relabel Method

Carlos Ontiveros and Bin Tang
Computer Science, California State University, Dominguez Hills

INTRODUCTION

Abstract:
Network flows studies how to move entities or objects, such as electrical currents, goods, vehicles, and network packets, from one point in a network to another point efficiently, while utilizing the underlying network resources cost-effectively. Coupling deep theoretical rigor and remarkable range of applicability, network flows spans over several broad disciplines including operations research, computer science, and engineering. In particular, the classic maximum flow problem finds the maximum amount of flow that can be sent from source node to sink node in a flow network, considering that edges in the flow network have capacities that restrain amount of flows on each edge. It has wide real world applications such as baseball elimination, airline scheduling, job scheduling, and network routing. The push-relabel algorithm is one of the most efficient algorithms for the maximum flow problem. Push operation and relabel operation are two basic operations used in the algorithm. In particular, during the execution of the algorithm, it maintains a "preflow" and gradually converts it into a maximum flow by moving flow locally between neighboring vertices using push operations under the guidance of an admissible network maintained by relabel operations. Cherkassky and Goldberg ("On Implementing Push-Relabel Method for the Maximum Flow Problem," Algorithmica 19 (1997), 390 -- 410) proposed an efficient implementation of the push-relabel algorithm for the maximum flow (available at http://www.avglab.com/andrew/soft.html). However, classic maximum flow fails to consider that different flow could have different weights or values. For example, in sensor networks, data collected by different sensors could have different values for the scientists to analyze the physical world. In this abstract, we study the maximum weighted flow problem, which is to maximize the total weight of flow in the network considering different flows have different weights. Maximum weighted flow is a generalization of the classic maximum flow problem, wherein each unit of flow has the same weight. We design an efficient optimal algorithm for maximum weighted flow problem. We implement our algorithm by modifying the implementation by Cherkassky and Goldberg and including the concept of the flow weights into the design and implementation. We generate different graph families using three graph generators available from DIMACS. Via extensive simulation, we show that it outperforms the push-relabel maximum flow algorithm in terms of the total preserved priorities.

MOTIVATION

This research was inspired by Dr. Andrew Goldberg's work in graph theory and optimization. He created the Push-Relabel Algorithm which aims to maximize the amount of entities or objects that can be moved from a source node to a terminal node.

For example, scientists must continuously monitor an active volcano and its environment to accurately assess its conditions. Scientific instruments are installed near the volcano and collect as much data as they can. Although many different types of data are collected, not all data is vital to know if a volcano is about to erupt. Due to energy depletion and storage constraints, not all data will be preserved. Consequently, scientists must somehow filter out unnecessary data and make sure the most vital data is preserved. In order to preserve the most vital data scientists must prioritize the data (flows) and give it a priority value (weight values). For example, data collected near a volcano may include: seismic data, temperature, infrasonic data, and wind speed. Different priority values will be given to each data: 4, seismic data; 3, temperature; 2, infrasonic data; 1, wind speed.

OBJECTIVES

- Analyze the Push-Relabel Algorithm which aims to maximize the amount of entities or objects that can be moved from the source node to a terminal node.
- Implement the new Maximum Weighted Flow Algorithm which aims to maximize the entities or objects with biggest priorities.

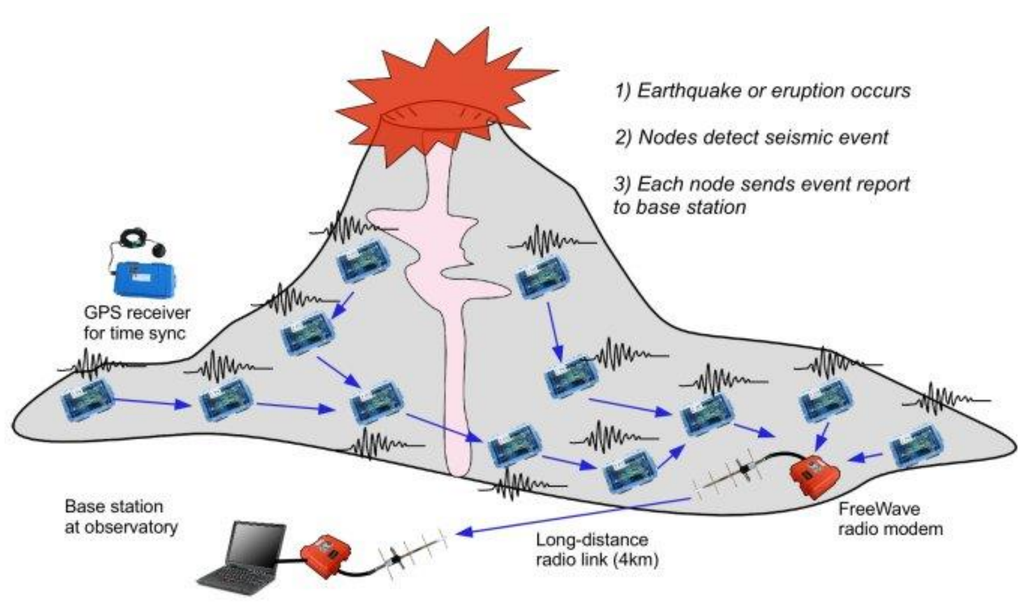
METHODS

Hi-Level Push-Relabel (HIPR) Program (Figure 2)

Algorithm: Push-Relabel Method
0. Notations:
$e(u)$: node u's excess flow
$h(u)$: node u's height
$cap(u,w)$: residual capacity of (u,w)
1. **if** $e(u) > 0$
2: **while** $(e(u) > 0$, there exists (u,w) s.t.
3. $h(u) = h(w) + 1$, and $cap(u,w) > 0)$
4. Push $y = \min\{e(u), cap(u,w)\}$ through
5. (u,w) by sending a message to w;
6: $e(u) = e(u) - y; e(w) = e(w) + y$;
7. update $cap(u,w)$;
8: **end while;**
9: **if** $e(u) > 0$
10: $h(u) = 1 + \min\{h(w): cap(u,w) > 0\}$;
11: Broadcast $h(u)$ to neighboring nodes;
12: **end if;**
13: **end if;**
15. **RETURN**

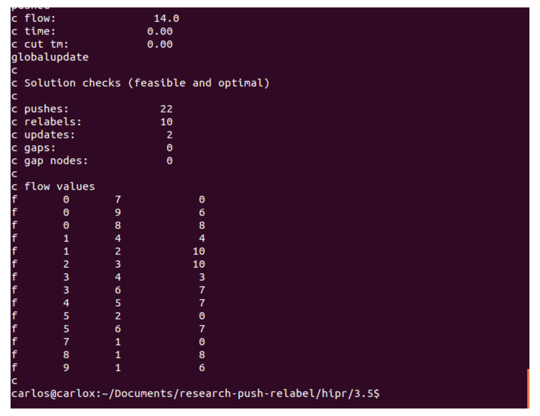
Algorithm: Maximum Weighted Flow
0. Notations:
$e(u)$: node u's excess flow
$h(u)$: node u's height
$cap(u,w)$: residual capacity of (u,w)
$b(u)$: node u's priority
Super source outputs weighted flows
$G' = (V', E), where\ E' = \{v_{b1}, v_{b2}, \ldots, v_{b3}\}$
Sort flows:
$b(u_1) \geq b(u_2) \geq \cdots \geq b(u_n)$
Input flows into directed graph based on highest priority

Figure 1



Source: http://fiji.eecs.harvard.edu/Volcano

Figure 2



RESULTS

Generated a weighted edge and weighted flow graph with these properties:
- Total Number of Nodes = 10
- Total Number of Edges = 14
- Number of different flows = 3
- Priority Value of Node 7 = 3
- Priority Value of Node 8 = 5
- Priority Value of Node 9 = 4

Figure 3 uses a standard algorithm to find the maximum flow which produces a total flow value of 54. In Figure 4 we used the Maximum Weighted Flow Algorithm that takes into consideration flows with priorities and produces the total value of 64 which is larger than the previous algorithm.

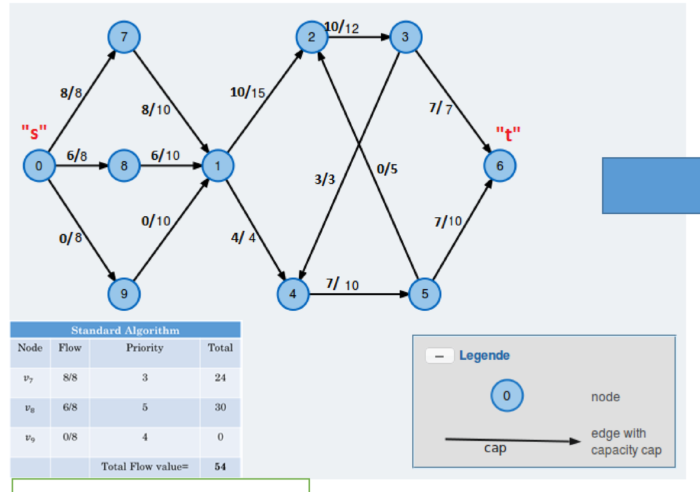


Figure 3

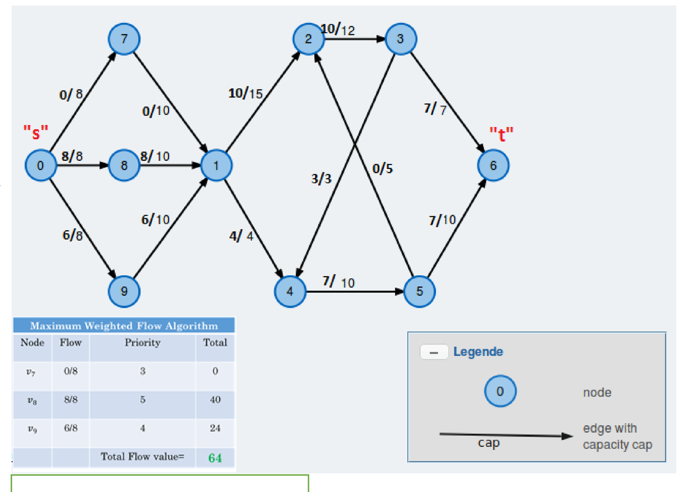


Figure 4

CONCLUSIONS

- Currently in the process of implementing our ideas
- Finalize the program
- Compare to unweighted flow Push-Relabel algorithm with priority value based algorithm to test performance
- Java Implementation

REFERENCES

- Xinyu Xue; Xiang Hou; Bin Tang; Bagai, R., (2013). "Data preservation in intermittently connected sensor networks with data priority," in Sensor, Mesh and Ad Hoc Communications and Networks IEEE (SECON)
- Cherkassky, Boris V.; Goldberg, Andrew V. (1995). "On implementing push-relabel method for the maximum flow problem"
- http://www.avglab.com/andrew/soft.html
- http://static.adrian-haarbach.de/idp-graph-algorithms/maxflow-push-relabel/index_en.html?graph=6

ACKNOWLEDGEMENTS

I would like to thank my professor and mentor, Dr. Bin Tang, for his guidance and patience throughout this research. I would also like to acknowledge the rest of the computer science department at Cal State Dominguez Hills. Finally I would like to thank the Louis Stokes Alliances for Minority Participation (LSAMP) program and the NSF for their continued support of our work.