ABSTRACT

THE COMPARISON OF *ENHANCED* SOLLIN ALGORITHM AND *ENHANCED* KRUSKAL ALGORITHM TO SOLVE THE MINIMUM ROUTING-COST SPANNING TREE (MRCST) PROBLEM

By

ASTRI REFORMASARI

A spanning tree that minimises the total pairwise distances between its vertices is the Minimum Routing Cost-Spanning Tree (MRCST) of a weighted graph. MRCST can be determined using several algorithms such as Sollin and Kruskal. In this study, the development (enhanced) of the Sollin and Kruskal algorithms was discussed to see whether the resulting MRCST can be more optimal and to see its effect and compare the results of the modification of the two algorithms. The data used was randomly generated data using a uniform distribution and the value was an integer. The data if represented in a graph, was a complete graph data that had a number of points of 10, 20, 30 to 100 with each number of nodes having 30 different problems. The results of the implementation using a python program show that the difference in MRCST values between the enhanced Sollin and enhanced Kruskal algorithms, compared with the original Sollin and Kruskal algorithms, tends to increase as graph order increases. For graph orders 40, 50, and 70–100, the enhanced Sollin algorithm produced lower MRCST values than the enhanced Kruskal algorithm. In contrast, the enhanced Kruskal algorithm performed better for graph orders 10–30 and 60.

Keywords: *MRCST, Sollin's Algorithm, Kruskal's Algorithm, Enhanced Sollin's Algorithm, Enhanced Kruskal's Algorithm.*

ABSTRAK

PERBANDINGAN ALGORITMA *ENHANCED* SOLLIN DAN ALGORITMA *ENHANCED* KRUSKAL UNTUK MENYELESAIKAN *MINIMUM ROUTING-COST SPANNING TREE (MRCST) PROBLEM*

Oleh

ASTRI REFORMASARI

Pohon merentang yang meminimumkan jumlah jarak berpasangan antara seluruh pasangan titiknya disebut sebagai *Minimum Routing-Cost Spanning Tree (MRCST)* dari suatu graf berbobot. MRCST dapat ditentukan dengan menggunakan beberapa algoritma seperti Sollin dan Kruskal. Pada penelitian ini, akan dibahas pengembangan (*enhanced)* algoritma Sollin dan Kruskal untuk melihat apakah MRCST yang dihasilkan dapat lebih optimal serta membandingkan hasil pengembangan kedua algoritma tersebut. Data dibangkitkan secara acak menggunakan distribusi *uniform* dan nilainya berupa bilangan bulat (*integer*). Data tersebut jika direpresentasikan dalam graf merupakan data graf lengkap yang memiliki banyaknya titik (*orde*) 10, 20, 30 sampai 100 dengan setiap *orde* memiliki 30 masalah berbeda. Hasil implementasi menggunakan program komputasi *Python* menunjukkan bahwa selisih nilai MRCST antara algoritma *enhanced* Sollin dan algoritma *enhanced* kruskal dengan algoritma Sollin dan algoritma Kruskal cenderung meningkat seiring bertambahnya *orde* graf. Pada *orde* 40, 50, dan 70 s.d. 100, hasil algoritma *enhanced* Sollin lebih unggul dibandingkan algoritma *enhanced* Kruskal sedangkan algoritma *enhanced* Kruskal unggul pada orde 10 s.d. 30 dan 60.

Kata Kunci: MRCST, Algoritma Sollin, Algoritma Kruskal, Algoritma *Enhanced* Sollin, Algoritma *Enhanced* Kruskal.