

ALGORITMA RUTE FUZZY TERPENDEK UNTUK KONEKSI SALURAN TELEPON

NELSA ANDRIANA, NARWEN, BUDI RUDIANTO

*Program Studi Matematika,
Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Andalas,
Kampus UNAND Limau Manis Padang, Indonesia,
nelsa.andriana@yahoo.com*

Abstrak. Salah satu permasalahan dalam graf yang cukup populer yang akan dibahas adalah masalah pencarian lintasan terpendek (*Shortest Path Problem*). Dalam mendefinisikan suatu masalah, terdapat cara untuk menemukan sirkuit terpendek yang dilihat dari setiap titik yang dilewati tepat satu kali yaitu dengan menggunakan beberapa algoritma. Namun pada penulisan ini akan dicari rute *fuzzy* terpendek dengan menggunakan Algoritma *Brute Force*, Algoritma Sisi Terurut, Algoritma Tetangga Terdekat (*Nearest Neighbor Algorithm*), dan Algoritma Kruskal yang diperumum pada data saluran telepon sebuah Perguruan Tinggi untuk interkom darat.

Kata Kunci: Himpunan *fuzzy*, sirkuit Hamilton, Algoritma *Brute Force*, Algoritma Sisi Terurut, Algoritma Tetangga Terdekat, Algoritma Kruskal yang diperumum

1. Pendahuluan

Permasalahan yang dapat dirumuskan dalam bentuk pencarian lintasan yang optimal diamati dalam arti sebuah angka kriteria yang telah ditetapkan. Kriteria tersebut sering dievaluasi dalam bentuk bobot yang memiliki nilai bilangan riil yang terkait dengan himpunan *fuzzy*. Teori himpunan *fuzzy* pertama kali diperkenalkan oleh Prof. Lotti A. Zadeh pada tahun 1965. Dalam teori himpunan *fuzzy* nilai keanggotaannya diperluas jangkauannya sehingga mencakup bilangan riil dalam interval $[0,1]$. Dalam mendefinisikan suatu masalah, terdapat cara untuk menemukan sirkuit terpendek yang dilihat dari setiap titik yang dilewati tepat satu kali yaitu dengan menggunakan beberapa algoritma.

2. Algoritma Rute *Fuzzy* Terpendek

Algoritma merupakan urutan logis langkah-langkah penyelesaian suatu masalah yang disusun secara sistematis. Suatu algoritma tidak saja harus benar, tetapi harus efisien. Kompleksitas dari suatu algoritma merupakan ukuran seberapa banyak komputasi yang dibutuhkan algoritma tersebut untuk menyelesaikan suatu masalah. Secara informal, algoritma yang dapat menyelesaikan suatu permasalahan dalam waktu singkat memiliki kompleksitas yang rendah, sementara algoritma yang membutuhkan waktu yang lama untuk menyelesaikan masalah mempunyai kompleksitas yang tinggi. Berikut merupakan beberapa algoritma yang digunakan.

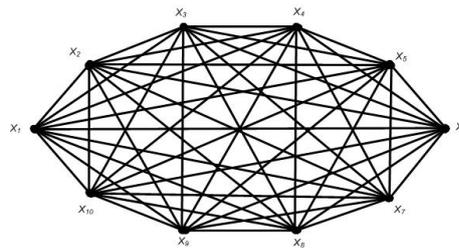
- (1) **Algoritma Tetangga Terdekat**, merupakan algoritma pencarian yang mengempatkan tetangga terdekat menurut beberapa fungsi jarak [2]. Langkah-langkah dari algoritma tetangga terdekat adalah sebagai berikut:
 - (a) Pilih sebuah titik untuk memulai.
 - (b) Dari titik yang dipilih tersebut, gerakkan ke titik terdekat yang belum pernah dilewati dengan bobot terkecil.
 - (c) Ulangi langkah (2), sehingga semua titik sudah dipilih dan hasil yang diperoleh adalah sirkuit lengkap.
- (2) **Algoritma Sisi Terurut**, merupakan pendekatan dengan mengurutkan sisi minimum yang sesuai dengan langkah-langkahnya [2]. Langkah-langkah algoritma Sisi Terurut adalah sebagai berikut:
 - (a) Pilih sisi yang memiliki bobot minimum pada graf.
 - (b) Ulangi langkah (1), kecuali:
 - Penambahan sisi akan membuat sirkuit tidak mengandung semua titik.
 - Penambahan sisi akan memberikan titik berderajat 3.
 - (c) Ulangi hingga membentuk sirkuit Hamilton *Fuzzy*.
- (3) **Algoritma Brute Force**, memecahkan masalah dengan sangat sederhana, langsung dan dengan cara yang jelas. Cara kerja algoritma *Brute Force* yaitu enumerasi (*list*) setiap solusi yang mungkin dengan cara sistematis, evaluasi setiap kemungkinan solusi "satu per satu" dan simpan solusi terbaik yang ditemukan, dan bila pencarian berakhir umumkan solusi terbaik [2]. Langkah-langkah algoritma *Brute Force* adalah sebagai berikut:
 - (a) Daftarkan semua kemungkinan sirkuit Hamilton *Fuzzy*
 - (b) Setelah itu, tambahkan bobot dari masing-masing sirkuit tersebut
 - (c) Pilih rangkaian sirkuit dengan bobot total minimal.
- (4) **Algoritma Kruskal yang diperumum**, untuk mencari *minimum spanning tree* dari graf berbobot [4]. Langkah-langkah algoritma Kruskal yang diperumum adalah sebagai berikut.
 - (a) Himpunan sisi dari graf diurutkan membesar sesuai bobot sisi tersebut.
 - (b) Buat *tree T* dengan memasukkan satu sisi terpendek dari graf tersebut.
 - (c) Ulangi langkah (2), (banyak sisi $T = \text{banyak titik graf} - 1$)
 - Pilih sisi selanjutnya dari graf.
 - Jika sisi tersebut tidak membuat sirkuit di T , tambahkan sisi tersebut ke T dan tambahkan titik-titik tersebut ke T .
 - (d) T adalah *minimum spanning tree* dari graf.
 - (e) Perhatikan titik-titik ujung (titik berderajat satu) dan titik yang berderajat ≥ 3 yang diperoleh.
 - (f) Hubungkan titik ujung tersebut ke titik lain sehingga:
 - Setiap titik tidak memiliki derajat ≥ 3 .
 - Terbentuk sirkuit Hamilton.
 - (g) Apabila setelah dihubungkan membentuk *cycle*, maka hapus sisi yang memiliki bobot paling besar diantara sisi yang membentuk *cycle* dengan

mengabaikan bobot sisi yang akan ditambahkan.

Dalam [6] telah ditunjukkan bahwa pada suatu graf lengkap K_n dengan n titik, $n \geq 3$, terdapat sebanyak $(n - 1)!/2$ buah sirkuit Hamilton. Pernyataan tersebut akan digunakan dalam pembahasan tulisan ini.

3. Algoritma Rute Fuzzy Terpendek untuk Koneksi Saluran Telepon

Suatu perguruan tinggi merencanakan sambungan saluran telepon untuk menghubungkan rute fuzzy yang efisien untuk interkom darat. Setiap departemen dimisalkan sebagai titik sebagai berikut : x_1 (ruang kantor), x_2 (departemen Matematika), x_3 (departemen Ekonomi), x_4 (departemen Sejarah), x_5 (departemen Ilmu Komputer), x_6 (Perpustakaan), x_7 (departemen Fisika), x_8 (departemen Kimia), x_9 (departemen Botani), dan x_{10} (departemen Pendidikan Fisik) (Gambar 1). Jarak



Gambar 1. Ilustrasi 10 titik departemen

di antara titik-titik tersebut direpresentasikan sebagai bobot fuzzy (lihat Tabel 1).

-	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
x_1	-	0,37	0,1	0,22	0,07	0,17	0,25	0,28	0,24	0,35
x_2	0,37	-	0,25	0,16	0,5	0,19	0,07	0,1	0,13	0,01
x_3	0,1	0,25	-	0,12	0,27	0,12	0,09	0,16	0,13	0,24
x_4	0,22	0,16	0,12	-	0,43	0,03	0,04	0,08	0,02	0,15
x_5	0,07	0,5	0,27	0,43	-	0,45	0,47	0,34	0,38	0,42
x_6	0,17	0,19	0,12	0,03	0,45	-	0,06	0,11	0,06	0,18
x_7	0,25	0,07	0,09	0,04	0,47	0,06	-	0,11	0,08	0,11
x_8	0,28	0,1	0,16	0,08	0,34	0,11	0,11	-	0,03	0,05
x_9	0,24	0,13	0,13	0,02	0,38	0,06	0,08	0,03	-	0,11
x_{10}	0,35	0,01	0,24	0,15	0,42	0,18	0,11	0,05	0,11	-

Tabel 1. Jarak antar departemen dengan nilai bobot Fuzzy

4. Hasil dan Pembahasan

Dengan menggunakan algoritma Tetangga Terdekat, titik yang dipilih x_2 (departemen Matematika). Dan selanjutnya penjelasan langkah-langkah dari algoritma Tetangga Terdekat untuk kasus ini dapat dilihat pada Tabel 2. sehingga sirkuit yang terbentuk adalah $x_2-x_{10}-x_8-x_9-x_4-x_6-x_7-x_3-x_1-x_5-x_2$ dengan bobot 0,96.

No.	(v_1)	Bobot Terkecil	(v_2)	Lintasan/Sirkuit yang Terbentuk	Jumlah Bobot
1	x_2	0,01	x_{10}	x_2-x_{10}	0,01
2	x_{10}	0,05	x_8	$x_2-x_{10}-x_8$	0,06
3	x_8	0,03	x_9	$x_2-x_{10}-x_8-x_9$	0,09
4	x_9	0,02	x_4	$x_2-x_{10}-x_8-x_9-x_4$	0,11
5	x_4	0,03	x_6	$x_2-x_{10}-x_8-x_9-x_4-x_6$	0,14
6	x_6	0,06	x_7	$x_2-x_{10}-x_8-x_9-x_4-x_6-x_7$	0,20
7	x_7	0,09	x_3	$x_2-x_{10}-x_8-x_9-x_4-x_6-x_7-x_3$	0,29
8	x_3	0,1	x_1	$x_2-x_{10}-x_8-x_9-x_4-x_6-x_7-x_3-x_1$	0,39
9	x_1	0,07	x_5	$x_2-x_{10}-x_8-x_9-x_4-x_6-x_7-x_3-x_1-x_5$	0,46
10	x_5	0,5	x_2	$x_2-x_{10}-x_8-x_9-x_4-x_6-x_7-x_3-x_1-x_5-x_2$	0,96

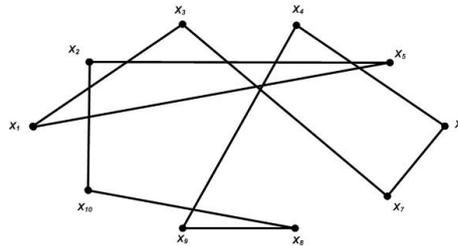
Tabel 2. Proses dengan menggunakan Algoritma Tetangga Terdekat

Akan ditentukan sirkuit terpendek untuk graf pada kasus ini dengan menggunakan algoritma sisi terurut. Berdasarkan Tabel 1, pilih sisi yang memiliki bobot minimum yaitu sisi $e_{2,10}$ dengan bobot 0,01. Lintasan yang diperoleh adalah x_2-x_{10} dengan bobot 0,01. Kemudian pilih sisi minimum selanjutnya yaitu $e_{10,8}$ dengan bobot 0,05, sehingga lintasan yang diperoleh adalah $x_2-x_{10}-x_8$ dengan bobot 0,06. Pilih sisi minimum selanjutnya yaitu $e_{8,9}$ dengan bobot 0,03, sehingga lintasan yang diperoleh adalah $x_2-x_{10}-x_8-x_9$ dengan bobot 0,09. Pilih sisi minimum selanjutnya yaitu $e_{9,4}$ dengan bobot 0,02, sehingga lintasan yang diperoleh adalah $x_2-x_{10}-x_8-x_9-x_4$ dengan bobot 0,11. Pilih sisi minimum selanjutnya yaitu $e_{4,6}$ dengan bobot 0,03, sehingga lintasan yang diperoleh adalah $x_2-x_{10}-x_8-x_9-x_4-x_6$ dengan bobot 0,14. Pilih sisi minimum selanjutnya yaitu $e_{6,7}$ dengan bobot 0,06, sehingga lintasan yang diperoleh adalah $x_2-x_{10}-x_8-x_9-x_4-x_6-x_7$ dengan bobot 0,2. Pilih sisi minimum selanjutnya yaitu $e_{7,3}$ dengan bobot 0,09, sehingga lintasan yang diperoleh adalah $x_2-x_{10}-x_8-x_9-x_4-x_6-x_7-x_3$ dengan bobot 0,29. Pilih sisi minimum selanjutnya yaitu $e_{3,1}$ dengan bobot 0,1, sehingga lintasan yang diperoleh adalah $x_2-x_{10}-x_8-x_9-x_4-x_6-x_7-x_3-x_1$ dengan bobot 0,39. Pilih sisi minimum selanjutnya yaitu $e_{1,5}$ dengan bobot 0,07, sehingga lintasan yang diperoleh adalah $x_2-x_{10}-x_8-x_9-x_4-x_6-x_7-x_3-x_1-x_5$ dengan bobot 0,46. Pilih sisi minimum selanjutnya yaitu $e_{5,2}$ dengan bobot 0,5, sehingga sirkuit yang diperoleh adalah $x_2-x_{10}-x_8-x_9-x_4-x_6-x_7-x_3-x_1-x_5-x_2$ dengan bobot 0,96. Karena sirkuit telah kembali ke titik awal, sehingga sirkuit yang terbentuk $x_2-x_{10}-x_8-x_9-x_4-x_6-x_7-x_3-x_1-x_5-x_2$ dengan bobot 0,96.

Sedangkan dengan menggunakan algoritma *brute force* akan ditentukan semua kemungkinan sirkuit Hamilton. Terdapat beberapa departemen yang akan direpresentasikan sebagai titik. Karena terdapat 10 departemen, maka terdapat 10 titik

yang akan membentuk sirkuit Hamilton. Dengan menggunakan Teorema sirkuit Hamilton, $n = 10$ dengan n adalah titik maka terdapat 181.440 kemungkinan sirkuit Hamilton. Untuk mendaftarkan semua kemungkinan sirkuit Hamilton tersebut akan membutuhkan banyak waktu dan ketelitian yang akurat.

Dengan menggunakan algoritma Kruskal yang diperumum diperoleh data yang diurutkan menurut bobot yang terkecil ke bobot yang terbesar. Dan apabila melanjutkan langkah-langkahnya, maka akan terbentuk sirkuit Hamilton seperti Gambar 2. Sehingga untuk algoritma Kruskal yang diperumum diperoleh $x_2-x_{10}-x_8-x_9-x_4-x_6-x_7-x_3-x_1-x_5-x_2$ dengan bobot 0,96.



Gambar 2. Sirkuit yang terbentuk dengan Algoritma Kruskal yang Diperumum

Berdasarkan hasil dari keempat algoritma, diperoleh bentuk sirkuit dengan jumlah bobot yang sama. Tetapi algoritma sisi terurut yang merupakan algoritma terbaik untuk kasus ini karena memiliki kompleksitas yang rendah dan ketelitian yang akurat sehingga menjadi lebih efisien.

5. Ucapan Terima kasih

Penulis mengucapkan terima kasih kepada Ibu Lyra Yulianti, Bapak Mahdhivan Syafwan, dan Bapak Zulakmal yang telah memberikan masukan dan saran sehingga paper ini dapat diselesaikan dengan baik.

Daftar Pustaka

- [1] Emut. Keterhubungan dan Planaritas Suatu Graf. Pengantar Teori Graf. diakses pada September 2013
- [2] Nirmala, G. and K. Uma. 2012. Fuzzy Shortest Route Algorithm for Telephone Line Connection. *International Journal of Scientific and Research Publications*. Volume 2
- [3] Bondy, J.A. and U.S.R.Murty. 1976. *Graph Theory with Applications*. Macmillan, London
- [4] Putro, H. Prihantoro. No Year. Prim Vs Kruskal, Perbandingan Algoritma Pencarian Pohon Merentang Minimum. *Journal of Mathematics*. Sekolah Teknik Elektro dan Informatika, Institut Teknologi Bandung. Bandung
- [5] Ratnasari, L.Y.D. Sumanto, dan T.A Novia. 2009. Komplemen Graf Fuzzy. *Journal of Scientific and Research Publications*. Universitas Diponegoro
- [6] Munir, R. 2011. *Matematika Diskrit Edisi Keempat*. Informatika, Bandung