

PENGGUNAAN ALGORITMA *RECURSIVE BEST FIRST SEARCH* DAN BAHASA PEMROGRAMAN PHP UNTUK MENYELESAIKAN *TRAVELING SALESMAN PROBLEM* (STUDI KASUS DI PT. ARTA BOGA CEMERLANG)

U. Windasari¹, A. Handayanto², A. N. Aini³

^{1,2,3}Pendidikan Matematika, Universitas PGRI Semarang, Jl. Dr. Cipto-Lontar No.1 Semarang
Email : ulfahwinda29@gmail.com¹, agunghandayanto@upgris.ac.id²,
auroranuraini@upgris.ac.id³

Abstrak

Pendistribusian barang dari suatu sumber ke beberapa tempat tujuan seringkali memiliki beberapa pilihan rute perjalanan. Masing-masing rute memiliki jarak tempuh yang berbeda dan berujung pada perbedaan biaya yang harus dikeluarkan. Berdasarkan hasil pengamatan di PT Arta Boga Cemerlang ditemukan permasalahan yaitu pendistribusian barang hanya dilakukan berdasarkan pengalaman. Padahal dengan menggunakan algoritma matematika, biaya distribusi barang dapat lebih dioptimalkan. Tujuan utama penelitian ini adalah menerapkan Algoritma *Recursive Best First Search* (RBFS) untuk menyelesaikan *Traveling Salesman problem* (TSP) di PT. Arta Boga Cemerlang. Pada penelitian ini juga digunakan Bahasa Pemrograman PHP untuk mempermudah perhitungan. Dengan menggunakan metode RBFS diperoleh panjang rute optimal yaitu 12.543 meter. Solusi ini memiliki selisih 2.059 meter dari rute yang digunakan oleh PT. Arta Boga Cemerlang sebelumnya.

Kata Kunci : Distribusi; TSP; Algoritma RBFS; PHP

Abstract

goods supply from a resource to some destinations have several route choices. each has different distances and cost. based on the observation at arta boga cemerlang inc, the writer knows that goods supply be done based on the experience. in this study, the writer uses mathematical algorithms to optimize the supplying cost. the objective of this study is to apply the algorithm of recursive best first search(rbfs) to solve traveling salesman problem (tsp) at boga cemerlang inc. in this study, the writer uses php programing language to easier the calculation. the result shows the optimal length route is 12.543 meters. it means there is a different length route between arta boga cemerlang inc and thartae calculation using algorithm of recursive best first search(rbfs). it is about 2.059 meters length.

Keywords: Distribution; TSP; RBFS Algorithm; PHP

PENDAHULUAN

Matematika adalah ilmu pengetahuan yang sangat penting karena akan selalu berkaitan dalam kehidupan sehari-hari (Suyitno, 2016:103). Misal pada sebuah perusahaan distributor yang perlu menyelesaikan permasalahan rute distribusi dan biaya transportasi menggunakan ilmu matematika. Kunci kesuksesan sebuah perusahaan distributor sangat berhubungan dengan bagaimana penentuan jalur yang digunakan supaya biaya transportasi efisien, produk sampai di

tujuan tepat waktu dan dalam kondisi baik (Aini, dkk, 2019).

Transportasi telah menjadi salah satu kebutuhan penting dalam kegiatan sehari-hari di kehidupan bermasyarakat. Letak suatu perusahaan distributor juga mempengaruhi proses distribusi karena akses transportasi dan jalur yang mudah dijangkau akan mempercepat proses pendistribusian barang (Siti, 2018).

Distribusi adalah kegiatan penyaluran hasil produksi berupa barang atau jasa dari produsen ke konsumen guna memenuhi kebutuhan manusia (Hariyono,

2012). Adapun permasalahan dalam kegiatan distribusi, pada umumnya bergantung pada letak, jarak dan berapa tempat barang tersebut akan didistribusikan oleh distributor. Pendistribusian barang dari sumber ke beberapa tujuan merupakan permasalahan yang cukup kompleks, karena akan menimbulkan beberapa jalur distribusi yang jarak dan waktunya akan semakin lama dan panjang (Muhammad, dkk, 2017). Bagi perusahaan distributor menentukan jalur menjadi hal yang sangat penting sebelum melaksanakan distribusi.

PT Arta Boga Cemerlang adalah perusahaan distributor yang berada di Jalan Jatibarang-Slawi KM7 RT 005/RW 009 Dukuhwaru, Tegal, Jawa Tengah. Perusahaan ini terbagi menjadi tiga divisi pemasaran, yaitu divisi *food & confectionaries (FC)* yang menjual produk makanan seperti wafer, biskuit, coklat, dan permen, divisi *personal care (PC)* yang menjual produk non makanan seperti pasta gigi, sikat gigi, *mouthwash*, dan divisi *sweet water plus (SW+)* yang menjual produk minuman seperti teh, jeli, air mineral, serta minuman kesehatan. Permasalahan dalam penelitian ini adalah banyaknya toko yang harus dikunjungi oleh setiap armada pengangkut barang tepat satu kali dan kembali lagi ke PT Arta Boga Cemerlang. Armada tersebut memiliki banyak pilihan rute perjalanan yang dapat dilalui, sehingga diperlukan sebuah perhitungan untuk memperoleh rute terpendek sehingga biaya transportasi menjadi minimal.

Menurut Faozi, dkk, (2017) *Traveling Salesman Problem* merupakan suatu optimasi dalam menyelesaikan suatu permasalahan yang dapat diterapkan pada berbagai kegiatan *routing* atau mekanisme pengiriman barang dari satu tempat ke tempat lainnya. Berdasarkan pengertian tersebut TSP dalam penelitian ini adalah setiap pengiriman barang yang dimulai dari kantor PT Arta Boga Cemerlang harus mengunjungi setiap toko tujuan tepat satu kali dan kembali lagi ke kantor PT Arta Boga Cemerlang. Terdapat banyak algoritma yang dapat digunakan dalam penyelesaian TSP, seperti algoritma genetika (GA), Ant Colony System (ACS), Evolutionary Programming (EP), RBFS dan algoritma lainnya.

Dalam penelitian ini algoritma yang diterapkan adalah Algoritma RBFS yang merupakan Algoritma Heuristik untuk menghasilkan *routing* atau pencarian dari suatu graf yang menghasilkan nilai optimal untuk menyelesaikan TSP (Faozi, 2016). Algoritma ini dapat menyelesaikan TSP dengan berbantu program bahasa pemrograman *Hypertext Preprocessor (PHP)*.

PHP merupakan bahasa pemrograman yang berbentuk *scripting*, dengan sistem kerja *interpreter* yang artinya *script* mentahnya tidak harus diubah kedalam *Source Code* yang disisipkan pada dokumen HTML (Nugroho, 2013:201; Rivai & Sukadi, 2013). TSP diselesaikan dengan menggunakan metode Algoritma RBFS untuk menghasilkan rute pendistribusian yang efektif dan optimal.

Menurut Munir, (2016:421-422) TSP termasuk ke dalam persoalan yang sangat terkenal dalam teori graf. Graf G didefinisikan pasangan himpunan (V, E) ditulis dengan notasi $G = (V, E)$, yang dalam hal ini V adalah himpunan tidak kosong dari simpul-simpul (*vertex* atau *node*) dan E adalah himpunan sisi (*edges* atau *arcs*) yang menghubungkan sepasang simpul. Sebuah algoritma tidak hanya harus benar, tetapi harus mangkus (efisien). Kemangkusan algoritma diukur dari berapa jumlah ruang dan waktu yang diperlukan untuk menjalankan, algoritma yang mangkus ialah yang meminimumkan kebutuhan ruang dan waktu (Munir, 2016:495-496).

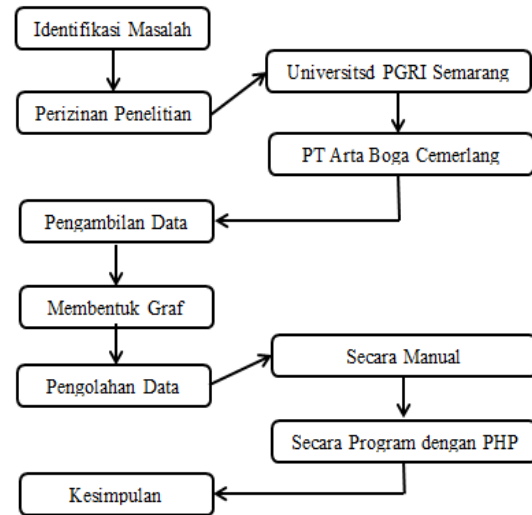
Adapun aplikasi yang digunakan untuk menentukan jarak antar titik yaitu *google maps* dan untuk memvisualisasikan peta menggunakan *google earth*.

Berdasarkan permasalahan yang ada, rencana pemecahan masalah adalah dengan mengubah permasalahan pendistribusian di PT Arta Boga Cemerlang ke dalam bentuk graf kemudian diselesaikan menggunakan metode Algoritma RBFS untuk mendapatkan rute pendistribusian yang optimal. Tujuan penelitian ini adalah menerapkan Algoritma RBFS dan berbantu program dengan PHP dalam penyelesaian TSP di PT Arta Boga Cemerlang.

METODE

Metode penelitian yang digunakan dalam penelitian ini adalah studi pustaka, observasi dan wawancara. Studi pustaka digunakan untuk memperoleh dan memahami teori-teori yang relevan dan berhubungan dengan topik atau masalah yang diteliti. Observasi yang dilakukan peneliti dengan melihat dan mengamati secara langsung untuk mencatat hal-hal yang diperlukan dalam penelitian. Metode wawancara yang dilakukan untuk mencari data atau informasi yang berkaitan dengan pendistribusian barang di PT Arta Boga Cemerlang.

Subjek penelitian adalah individu, benda, atau organisme yang dijadikan sumber informasi yang dibutuhkan dalam pengumpulan data penelitian. Alur penelitian yang dilakukan peneliti dapat dilihat pada Bagan 1 berikut:



Bagan 1 Alur Penelitian

HASIL DAN PEMBAHASAN

Algoritma RBFS merupakan salah satu metode yang digunakan untuk menyelesaikan TSP. Proses perhitungan metode Algoritma RBFS dilakukan dengan membandingkan bobot masing-masing *node* yang saling terhubung, kemudian dicari bobot terpendeknya.

Tabel 1 berikut menunjukkan data pengiriman barang berdasarkan titik terjauh dari 38 agen yang harus dikunjungi oleh sebuah armada pengangkut barang pada bulan September 2019 di PT Arta Boga Cemerlang.

Tabel 1. Titik terjauh yang harus dilewati

No	Node	Kode	Tempat	Lokasi
1	v_1	A	PT Arta Boga Cemerlang	Jalan Jatibarang-Slawi KM7 RT 005/RW 009.
2	v_2	B	Toko One Josh	Desa Selapura Selatan Gang Setiakawan Jatibarang
3	v_3	C	Toko Pak Meri	Jl Raya Selapura Jatibarang
4	v_4	D	Toko Wulan	Jl Raya Kebonagung
5	v_5	E	Toko Qilla	Gangg Umrik Selapura Jatibarang
6	v_6	F	Toko Rejeki	Jl Raya Timur Jatibarang
7	v_7	G	Toko Fatonah	Ds. Buaran 2 Timur Pertigaan Brebes
8	v_8	H	Toko Rosidah	Kios Samping Pasar No 14
9	v_9	I	Toko Harum	Jl Pramuka Rt 15 Rw 2
10	v_{10}	J	Toko Bagus	Jl Dukuhmaja Jatibarang Brebes
11	v_{11}	K	Toko Lili Berkah	Komlek Pasar Klampis Jatibarang

Berdasarkan hasil wawancara dengan sopir armada jarak tempuh pendistribusian

barang yang biasa dilakukan tanpa menggunakan Algoritma RBFS mengacu

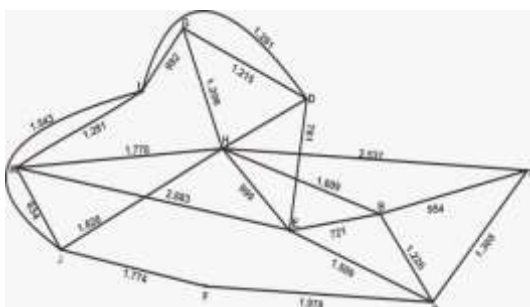
pada Tabel 1 adalah PT Arta Boga Cemerlang, Toko One Josh, Toko Qilla, Toko Wulan, Toko Fatonah, Toko Rosidah, Toko Lili Berkah, Toko Harum Snack, Toko Bagus, Toko Rejeki, Toko Meri dan kembali lagi ke PT Arta Boga Cemerlang sejauh 14.602 meter. Data lokasi agen digunakan

untuk Penentuan jarak dari PT Arta Boga Cemerlang ke agen, maupun dari agen ke agen. Data keterhubungan antar *node* ditampilkan pada Tabel 2.

Tabel 2. Bobot *node* yang berhubungan

Titik terbuhung	Bobot (Meter)
A-B	956
A-C	1305
A-H	2537
B-C	1226
B-E	721
B-H	1609
C-E	1509
C-F	1978
D-E	781
D-G	1281
D-H	727
D-I	1215
E-H	999
E-K	2683
F-J	1774
G-H	1206
G-I	982
H-J	1820
H-K	1778
I-J	1543
I-K	1281
J-K	834

Penerapan Algoritma RBFS dalam penyelesaian TSP di PT Arta Boga Cemerlang memodelkan lokasi baik perusahaan maupun agen sebagai himpunan titik, sedangkan jarak antar lokasi sebagai himpunan sisi dan panjang jarak sisi tersebut yang disebut bobot (Tanjung & Sopiah, 2010), sehingga keadaan yang terjadi dapat digambarkan dalam sebuah graf seperti ditampilkan pada Gambar 1.



Gambar 1. Graf bobot (meter)

Penerapan Algoritma Recursive Best First Search (RBFS) dalam penyelesaian Traveling Salesman Problem

Penerapan Algoritma RBFS dalam penyelesaian TSP menurut (Faozi, 2017:215) adalah sebagai berikut :

Tahap iterasi pertama menentukan lintasan yang dapat dilalui dari titik awal

yaitu A dapat dilihat pada Tabel 3. Berdasarkan Gambar 1 lintasan yang mungkin dilalui oleh titik A adalah A-B sejauh 954 meter, A-C sejauh 1.305 meter, dan A-H sejauh 2.537 meter. Nilai

minimum dari ketiga jalur tersebut adalah A-B sejauh 954 meter, sedangkan *threshold* atau nilai minimum kedua pada jalur A-C sejauh 1.3035 meter.

Tabel 3. Langkah Pertama Penentuan Solusi Optimum

No	Open	Rute	Minimum	Threshold
1	A (0)	A-C (1305 m) A-B (954 m) A-H (2537 m)	A-B (954 m)	A-C (1305 m)

Tahap iterasi kedua yaitu menentukan lintasan yang terdapat pada jalur Minimum yaitu A-B. Berdasarkan Gambar 1 jalur yang dapat dilalui yaitu A-B-C sejauh 2.180 meter, A-B-E sejauh 1.675 meter, dan A-B-H sejauh 2.563 meter dapat dilihat pada Tabel 4.

Langkah minimum selanjutnya yaitu dengan membandingkan nilai *threshold* yang sudah didapat dengan nilai rute yang telah dicari. Berdasarkan Tabel 4 nilai minimum yaitu jalur A-C sejauh 1.305 meter, sedangkan *threshold* pada jalur A-B-E sejauh 1.675 meter.

Tabel 4. Langkah Kedua Penentuan Solusi Optimum

No	Open	Rute	Minimum	Threshold
1	A (0)	A-C (1305 m) A-B (954 m) A-H (2537 m)	A-B (954 m)	A-C (1305 m)
2	A-B (954 m)	A-B-C (2180m) A-B-E (1675 m) A-B-H (2563m)	A-C (1305 m)	A-B-E (1675 m)

Tahap ketiga yaitu dengan memilih jalur *threshold* pada langkah pertama atau nilai di kolom Minimum pada Langkah Kedua yaitu jalur A-C sejauh 1.305

meter. Sehingga nilai minimumnya adalah A-B-E sejauh 1.675 meter dan A-B-C sejauh 2.180 meter sebagai *threshold* pada Tabel 5.

Tabel 5. Langkah Ketiga Penentuan Solusi Optimum

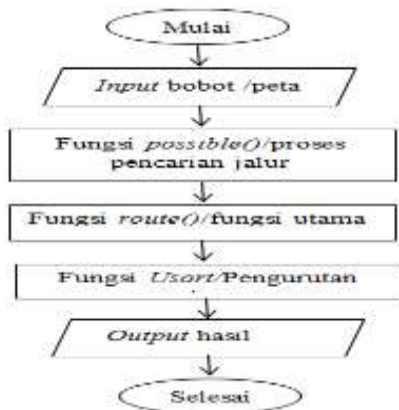
No	Open	Rute	Minimum	Threshold
1	A (0 m)	A-C (1305 m) A-B (954 m) A-H (2537 m)	A-B (954 m)	A-C (1305 m)
2	A-B (954 m)	A-B-C (2180m) A-B-E (1675 m) A-B-H (2563 m)	A-C (1305 m)	A-B-E (1675 m)
3	A-C (105 m)	A-C-B (2531m) A-C-E (2814 m) A-C-F (3283 m)	A-B-E (1675 m)	A-B-C (2180 m)

Tahap keempat yaitu diberikan bobot ∞ pada jalur yang sudah melalui setengah dari

jumlah titik yang harus dikunjungi dan jika diprediksi tidak memenuhi TSP. Contoh jalur A-C-E-H-B dapat dilihat pada Gambar 1 diketahui sejauh 5.422 meter, jika diteruskan maka tidak ada lagi jalur yang dapat dilalui karena seluruh jalur yang mungkin dapat dilalui di titik B sudah dilalui sedangkan titik

Pembuatan Program Algoritma Recursive Best First Search untuk Menyelesaikan Traveling Salesman Problem dengan PHP

Pembuatan program diawali dengan pembuatan *Backend* dan *Frontend*. *Backend* adalah aplikasi sisi server yang berisi konfigurasi dasar dari suatu aplikasi seperti *update*, *delete*, *create*, *read*, koneksi, algoritma RBFS, dan sebagainya. Sedangkan *Frontend* merupakan aplikasi sisi *client* yang berisi konfigurasi sisi *client controller* dan menu yang ditampilkan ke *user* seperti menu *home*, menu keadaan, menu solusi dan menu *about*. Bagan 2 merupakan *Flowchart* dari program Algoritma Recursive Best First Search.



Bagan 2. *Flowchart* program algoritma RBFS

Langkah awal dalam pembuatan program ini adalah menentukan pemetaan lokasi dan bobot antar lokasi. Menu keadaan merupakan menu yang menampilkan keadaan graf atau peta dan penginputan keadaan (titik dan sisi). Peta menggambarkan wilayah Brebes beserta lokasi PT Arta Boga Cemerlang dan lokasi agen beserta bobot antar

yang lain belum teralui, ini artinya jalur tersebut tidak dapat memenuhi TSP.

Tahap kelima yaitu menentukan rute yang memuat siklus Hamilton. Rute tersebut adalah A-B-E-D-H-G-I-K-J-F-C-A atau sebaliknya yaitu A-C-F-J-K-I-G-H-D-E-B-A dengan bobot 12.543 meter.

agen. Peta dibuat dalam bentuk *array* objek dimana masing-masing agen berisi ID agen, kode agen, jalur yang mungkin dilalui sekaligus bobot antar agen. *Script* peta dapat dilihat pada Gambar 2.

```

<?php
$peta = array();

$peta = array(
    'a' => (object) array(
        'id' => 'a',
        'name' => 'A',
        'path' => array(
            'h' => 2537
        )
    ),
    'b' => (object) array(
        'id' => 'b',
        'name' => 'B',
        'path' => array(
            'a' => 954,
            'h' => 1609
        )
    ),
);
  
```

Gambar 2. Representasi peta dalam bahasa pemrograman

Menentukan fungsi *possible()* atau proses pencarian jalur melalui menu solusi. Menu ini merupakan menu yang menampilkan rute dan bobot dari graf atau keadaan yang dicari. Dalam proses pencarian jalur, peta tersebut 'dibaca' dengan fungsi *possible()* untuk mengetahui kemungkinan jalur yang akan dilalui dari suatu lokasi. Fungsi *possible()* tampak pada Gambar 3 dengan inputnya id agen, dan outputnya adalah daftar agen yang mungkin dilalui.

```

function possible($agen){
    $ret = array();
    foreach($this->petas as $next){
        foreach($next->path as $key=>$target){
            $tmp = array();
            if($next->id == $agen){
                $tmp['id'] = $key;
                $tmp['len'] = $target;
                array_push($ret,$tmp);
            }
            else{
                if($key == $agen){
                    $tmp['id'] = $next->id;
                    $tmp['len'] = $next->path[$agen];
                    array_push($ret,$tmp);
                }
            }
        }
    }
    return $ret;
}

```

Gambar 3. Fungsi *possible()*

Menentukan fungsi *route()* atau fungsi utama membutuhkan masukan agen yang sedang dibuka (*open*) dan menghasilkan jalur dengan bobot (*cost*) minimal dan jalur yang akan dijadikan sebagai *threshold/backup*. *Source code* fungsi *route()* dapat dilihat pada Gambar 4. Proses fungsi *route()* diawali dengan mencari rute yang mungkin dilewati dengan memanggil fungsi *possible()*, kemudian daftar jalur yang dilewati akan dihitung satu-persatu dengan perulangan. Variabel *\$tmp[*len*min]* berfungsi untuk menghitung besar bobot masing-masing jalur dari jalur awal.

```

function route($start){
    $next = $this->possible($start['id']);

    $lastone = true;
    foreach($next as $next){

        $chopath = $start['path'].'-'.$next['id'];
        $chokp = explode('-', $chopath);

        if(isset($petas[$start['path'],$next['id']])){
            ($chokp[0]==$start['id'] && sizeof($chokp)==sizeof($this->petas)+1){
                $tmp = $next;
                if($start['path']==''){
                    $tmp['lenmin'] = $next['len'];
                    $tmp['path'] = $start['id'].'-'.$tmp['id'];
                }
                else{
                    $tmp['lenmin'] = $start['lenmin'] + $next['len'];
                    $tmp['path'] = $start['path'].'-'.$tmp['id'];
                }
            }
            array_push($this->routing,$tmp);
            $lastone = false;
        }
    }
}

```

Gambar 4. Fungsi *route()*

Menentukan fungsi *usort()* atau pengurutan berdasarkan bobot terkecil. Setelah diurutkan, proses selanjutnya adalah jalur dengan bobot terkecil akan dibuka (variabel *\$open*) dan jalur dengan bobot terkecil kedua akan

dijadikan sebagai *threshold* untuk perbandingan (variabel *back*). Kemudian pada bagian *\$this->route(\$open)* akan membuka jalur baru. Pada fungsi *AGEN()* dengan masukan jalur yang terbuka, fungsi *AGEN()* melakukan penerjemahan dari kode menjadi nama agen, sehingga pada keluaran program akhir berupa urutan nama agen yang dilewati (Sikel Hamilton). *Source code* fungsi *usort()* dapat dilihat pada Gambar 5.

```

usort($this->routing,function($a,$b){ return $a['lenmin'] - $b['lenmin']; });

$open = $this->routing[0];
$back = $this->routing[1];

if($open['path']==''){
    $tmp = array();
    $tmp['path'] = $this->AGEN($open['path']);
    $tmp['cost'] = $open['lenmin'];
    array_push($this->result,$tmp);
}

if($open['path']!=''){
    unset($this->routing[0]);
    $this->route($open);
}

function AGEN($path){
    foreach($this->petas as $agen){
        $path = str_replace($agen->id,$agen->name,$path);
    }
    return $path;
}

```

Gambar 5. Fungsi *usort()*

Setelah konfigurasi selesai, peneliti mengaktifkan aplikasi *xampp control* sebagai server penerjemah PHP. Kemudian memanggil program dalam *web browser* diperoleh jalur terpendek yang memenuhi TSP pada rute pendistribusian di PT Arta Boga Cemerlang adalah A-B-E-D-H-G-I-K-J-F-C-A dengan jarak 12.543 meter.

Dengan metode ini, rute pendistribusian yang diperoleh adalah dari PT Arta Boga Cemerlang - Toko One Josh - Toko Qilla - Toko Wulan - Toko Rosidah - Toko Fatonah - Toko Harum Snack - Toko Lili Berkah - Toko Bagus - Toko Rejeki - Toko Meri - PT Arta Boga Cemerlang dengan jarak 12.534 meter. Gambar 6 adalah visualisasi rute hasil perhitungan menggunakan *google earth*, dimana jarak antar titik hanya digambarkan menggunakan garis lurus. Hal tersebut tidak mempengaruhi hasil karena

dalam menentukan jarak peneliti menggunakan *google maps*.



Gambar 6. Hasil rute TSP menggunakan algoritma RBFS

Rute pada Gambar 6 lebih optimal dengan selisih 2.059 meter dibandingkan dengan rute sebelumnya tanpa menggunakan Algoritma RBFS, dan program Algoritma ini menurut Munir, (2016:496-497) mangkus karena mampu meminimumkan kebutuhan ruang dan waktu.

SIMPULAN

Berdasarkan hasil pembahasan penerapan metode Algoritma RBFS untuk menyelesaikan TSP, dapat ditarik kesimpulan bahwa Algoritma RBFS dapat diterapkan untuk menyelesaikan TSP di PT Arta Boga Cemerlang. Hasil perhitungan yang diperoleh berupa siklus Hamilton yang memenuhi TSP dengan jarak terpendek.

DAFTAR PUSTAKA

- Aini, A. N., Sari, G., & Shodiqin, A. (2019). Aplikasi Metode Lowest Supply Lowest Cost (LSLC) Pada Masalah Transportasi Tidak Seimbang (Studi Kasus Pada Distribusi Garam Ud. Aditya Mandiri). *Jurnal Ilmiah Teknosains*, Volume V No 1 (28-34).
- Faozi. (2016). Penerapan algoritma *recursive best first search* dalam penyelesaian *traveling salesman*

problem di PT. Bintang Service Management. Matematika. Fakultas Matematika dan Ilmu Pengetahuan Alam. Universitas Negeri Semarang.

- Faozi, F., Suyitno, A., & Arifudin, R. (2017). Penerapan algoritma *recursive best first search* (RBFS) dalam penyelesaian *traveling salesman problem* (TSP) di PT. Bintang service management. *Unnes Journal of Mathematics*. 6(2), 211–220.

- Hariyono, Ahmad. (2012). Analisis penerapan transportasi distribusi dengan menggunakan NWCM dan SSM pada harian Tribun Timur Makassar. Manajemen. Fakultas Ekonomi dan Bisnis. Universitas Hasanuddin Makassar.

- Muhammad, Bakhtiar, & Rahmi, M. (2017). Penentuan Rute Transportasi Distribusi Sirup Untuk Meminimalkan Biaya. *Industrial Engineering J Journal*, 6(1), 10–15.

- Munir. 2016. *Matematika diskrit*. Bandung : Informatika.

- Nugroho, B. 2013. *Aplikasi pemrograman web dinamis dengan PHP dan MySQL*. Yogyakarta : Gaya Media

- Pramesti, A., Novaliendry, D., Sriwahyuni, T., Studi, P., Teknik, P., Teknik, F., & Negeri, U. (2014). Perancangan Website E-Commerce Express Order System for Reseller Dropshipper Menggunakan Hypertext Preprocessor. *Jurnal Vokasional Teknik Elektronika & Informatika*, 2(2).

- Rani, S. (2016). Pemanfaatan Google Maps Api Untuk Visualisasi Data Base Transceiver Station. *Teknoin*, 22(2), 1–9.

- Rivai, D. A., & Sukadi. (2013).

- Pembuatan Website Profil Sekolah Menengah Kejuruan (Smk) Miftahul Huda Ngadirojo. *IJNS-Indonesian Journal on Networking and Security*, 2(3), 14–18.
- Sheppard, S. R. J., & Cizek, P. (2009). The ethics of Google Earth: Crossing thresholds from spatial data to landscape visualisation. *Journal of Environmental Management*, 90(6), 2102–2117.
- Siti, A. (2018). Transportasi Publik dan Aksesibilitas Masyarakat Perkotaan. *Jurnal Teknik Sipil*, 19(2), 90–96.
- Suyitno. 2018. *Filsafat matematika*. Yogyakarta : Magnum Pustaka Utama.
- Tanjung, W. N. & Sopiah. 2010. *Implementing Ant Colony Optimization (Aco) In Traveling Salesman Problem (Case Study at Distribution Store PB)*. Jakarta, Proceeding 7th International Seminar on Industrial Engineering and Management.