

DECISION SUPPORT SYSTEM TOOL UNTUK PENYELESAIAN PERMASALAHAN LINEAR BERBASIS SIMPLEX DAN REVISED SIMPLEX

Arya Tandy Hermawan, Gunawan, Yudhi Christian Mahono

Jurusan Teknik Informatika, Sekolah Tinggi Teknik Surabaya

Jl. Ngagel Jaya Tengah 73-77, Surabaya 60284, Indonesia; Telp. (031) 5027920, Fax. (031) 5041509

E-mail: arya@stts.ac.id, admin@hansmichael.com, yoodee_77@yahoo.com

ABSTRAK

Pada permasalahan optimasi sumber daya, baik itu minimalisasi biaya ataupun maksimalisasi keuntungan, yang bersifat linear, dapat dibuat sebuah tool yang berguna untuk membantu menyelesaikan permasalahan tersebut sehingga didapat hasil kombinasi sumber daya yang optimal. Sistem ini dirancang untuk dapat menangani berbagai bidang permasalahan linier karena sifatnya yang terbuka. DSS tool ini akan menggunakan metode penyelesaian simplex atau revised simplex dalam membantu penyelesaian permasalahannya. Namun sebelum menerapkan permasalahan linear yang dihadapi pada tool yang telah dibuat, terlebih dahulu perlu untuk melakukan pembentukan model matematika dari permasalahan tersebut. Dengan memanfaatkan DSS tool ini, maka semua permasalahan linear akan dapat diselesaikan dengan tepat dan cepat.

Kata Kunci: dss tool, linear programming, simplex, revised simplex, optimasi

1. PENDAHULUAN

Keputusan harus dibuat karena ada beberapa alternatif yang dapat dipilih untuk mencapai tujuan yang ditetapkan. Model pengambilan keputusan merupakan alat yang menggambarkan permasalahan keputusan sedemikian rupa sehingga memungkinkan identifikasi dan evaluasi sistematis semua alternatif keputusan yang tersedia.

Proses pengoptimalan dimulai dengan pengamatan yang mendalam dan formulasi masalah, lalu diikuti dengan pembentukan model ilmiah (khususnya model matematika) yang menggambarkan inti sistem nyata.

Optimasi diperlukan karena keterbatasan sumber daya yang dimiliki. Optimasi adalah proses pencarian solusi terbaik, tidak selalu keuntungan paling tinggi yang bisa dicapai jika tujuan pengoptimalan adalah memaksimalkan keuntungan, atau tidak selalu biaya paling kecil yang bisa ditekan jika tujuan pengoptimalan adalah meminimumkan biaya produksi. Tiga elemen penting yang harus diidentifikasi, yaitu tujuan, alternatif keputusan dan sumber daya yang membatasi.

Tujuan berbentuk memaksimalkan atau meminimumkan. Proses maksimalisasi digunakan jika tujuan pengoptimalan berhubungan dengan keuntungan, penerimaan, dan sejenisnya. Proses minimalisasi dipilih jika tujuan pengoptimalan berhubungan dengan biaya, jarak, dan sejenisnya.

Keputusan harus diambil untuk alternatif keputusan yang disediakan, pengambilan keputusan dihadapkan pada beberapa pilihan untuk mencapai tujuan. Alternatif keputusan menggunakan sumber daya terbatas yang dimiliki pengambil keputusan. Keterbatasan sumber daya inilah yang mengakibatkan dibutuhkan proses optimasi.

2. DISIPLIN TERKAIT

Terdapat dua teori yang mendukung pembuatan DSS tool ini. Berikut ini akan dijelaskan secara singkat mengenai *decision support system* dan *linear programming*.

2.1 Decision Support System

Definisi awal decision support system (DSS) adalah suatu sistem yang ditujukan untuk mendukung manajemen pengambilan keputusan. Sistem berbasis model yang terdiri dari prosedur-prosedur dalam pemrosesan data dan pertimbangannya untuk membantu user dalam mengambil keputusan. Agar berhasil mencapai tujuannya maka sistem tersebut harus: sederhana, kuat, mudah untuk dikontrol, mudah beradaptasi, lengkap pada hal-hal penting, dan mudah untuk digunakan. Tiga buah tujuan DSS yang harus dicapai adalah: membantu user membuat keputusan, mendukung penilaian user bukan mencoba untuk menggantikannya, meningkatkan efektifitas pengambilan keputusan user.

DSS tool yang dikembangkan ini merupakan sebuah tool yang berfungsi untuk mengatasi semua jenis permasalahan (bersifat terbuka) yang sifatnya linier, artinya DSS tool ini harus didesain sefleksibel mungkin agar dapat menangani berbagai bidang permasalahan. Untuk itu pada tahap awal pengembangan tool ini, langkah awal yang perlu dilakukan adalah mengumpulkan berbagai jenis permasalahan linier yang dipandang mampu untuk diselesaikan dengan baik oleh tool ini. Kemudian dirancang desain yang tepat dan tidak kompleks namun dapat mewakili dan menyelesaikan semua permasalahan tersebut.

Sistem ini dikembangkan untuk meningkatkan efektifitas dari pengambilan keputusan, lebih

daripada efisiensinya. Serta bagian terpenting dari sebuah karakteristik DSS adalah secara khusus ditujukan untuk mendukung dan bukan untuk menggantikan peranan dari pengambil keputusan.

2.2 Linear Programming

Linear programming merupakan suatu metode matematika yang dapat digunakan untuk membantu dalam perencanaan dan pengambilan keputusan dalam penggunaan sumber daya yang terbatas di tengah-tengah aktivitas-aktivitas yang saling bersaing melalui jalan atau cara yang terbaik. Pemrograman linier meliputi perencanaan aktivitas untuk mendapatkan hasil optimal, yaitu sebuah hasil yang mencapai tujuan terbaik (menurut model matematika) diantara semua kemungkinan alternatif yang ada.

Karakteristik-karakteristik pada linear programming adalah: fungsi tujuan (untuk memaksimalkan atau meminimumkan sesuatu), fungsi pembatas yang membatasi tingkatan pencapaian tujuan, adanya beberapa alternatif tindakan yang bisa dipilih, fungsi tujuan dan kendala dalam permasalahan diekspresikan dalam bentuk persamaan atau pertidaksamaan linier

Metode *simplex* adalah suatu metode yang secara sistematis dimulai dari suatu pemecahan dasar yang dimungkinkan ke pemecahan dasar yang lainnya dan ini dilakukan tahap demi tahap yang disebut dengan iterasi (dengan jumlah iterasi yang terbatas) sehingga pada akhirnya akan tercapai sesuatu pemecahan dasar yang optimum dan setiap langkah menghasilkan suatu nilai dari fungsi tujuan yang selalu lebih optimal atau sama dari langkah-langkah sebelumnya.

Metode dua fase digunakan jika variabel basis awal terdiri dari variabel buatan. Disebut sebagai metode dua fase karena proses optimasi dilakukan dalam dua tahap. Pada tahap pertama merupakan proses optimasi variabel buatan, sedangkan pada tahap kedua proses optimasi variabel keputusan.

Revised simplex method merupakan metode dari linear programming yang menggunakan bentuk matrix dalam prosesnya. Dengan bantuan matrix, waktu perhitungan bisa dipersingkat dan mempermudah kerumitan perhitungan metode *simplex* biasa. Menggunakan metode *simplex* yang telah direvisi dalam pemrograman akan mampu mengefisienkan memori dan total waktu perhitungan.

3. IMPLEMENTASI

Implementasi sistem ini terbagi ke dalam dua tahap. Tahap pertama adalah tahap pembentukan model matematika permasalahan linier yang bertujuan untuk menyiapkan segala data yang diperlukan dalam tahap selanjutnya dan tahap kedua adalah iterasi pencarian hasil optimal. Berikut ini dijelaskan detail dari masing-masing tahap.

3.1 Pembentukan Model Matematika

Model matematika merupakan representasi kuantitatif tujuan dan sumber daya yang membatasi sebagai fungsi variabel keputusan. Model matematika permasalahan optimasi terdiri dari dua bagian. Bagian pertama memodelkan tujuan optimasi. Model matematika tujuan selalu menggunakan bentuk persamaan. Bentuk persamaan digunakan karena nantinya diinginkan mendapatkan solusi optimum pada satu titik. Fungsi tujuan yang akan dioptimalkan hanya satu; bukan berarti bahwa permasalahan optimasi hanya dihadapkan pada satu tujuan. Tujuan dari suatu usaha bisa lebih dari satu.

Bagian kedua merupakan model matematika yang merepresentasikan sumber daya yang membatasi. Fungsi pembatas bisa berupa persamaan ($=$) atau pertidaksamaan (\leq atau \geq). Fungsi pembatas disebut juga sebagai *constraint*. Konstanta dalam fungsi pembatas maupun pada tujuan disebut sebagai parameter model. Model matematika dapat menggambarkan suatu permasalahan secara lebih ringkas sehingga permasalahan akan lebih mudah untuk dipahami. Model matematika mampu menjembatani penggunaan teknik matematika dan teknologi komputer untuk menganalisis permasalahan. Namun, model matematika juga memiliki kelemahan yaitu, tidak semua karakteristik sistem dapat dengan mudah dimodelkan menggunakan fungsi matematika. Meskipun dapat dimodelkan dengan fungsi matematika, terkadang penyelesaiannya susah diperoleh karena kompleksitas fungsi dan teknik yang dibutuhkan.

Bentuk umum pemrograman linier yaitu:

Fungsi tujuan: Maksimumkan atau Minimumkan

$$Z = c_1x_1 + c_2x_2 + \dots + c_nx_n$$

Sumber daya yang membatasi (kendala):

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = / \leq / \geq b_1$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = / \leq / \geq b_2$$

:

$$a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = / \leq / \geq b_m$$

$$x_1, x_2, \dots, x_n \geq 0$$

x_1, x_2, \dots, x_n : variabel keputusan.

c_1, c_2, \dots, c_n : kontribusi masing-masing variabel keputusan terhadap tujuan.

$a_{11}, \dots, a_{1n}, \dots, a_{mn}$: penggunaan per unit variabel keputusan akan sumber daya yang membatasi.

b_1, b_2, \dots, b_m : jumlah tiap sumber daya yang ada.

$x_1, x_2, \dots, x_n \geq 0$: batasan nonnegatif.

3.2 Pencarian dan Pembacaan Hasil Optimal

DSS tool ini memanfaatkan salah satu dari metode penyelesaian linear programming yang dapat dipilih yaitu, *simplex* method atau *revised simplex* method dalam pencarian hasil optimalnya. Setiap metode akan membantu untuk memberikan hasil akhir yang optimal. Perbedaan dari kedua metode di atas adalah pada proses pencarian komposisi optimalnya. Sebelum memasuki tahap iterasi pencarian hasil optimal, selalu diawali dengan tahap

pemeriksaan optimalitas. Pada tahap ini bisa menghasilkan tiga buah kemungkinan yaitu: tidak ada solusi yang dapat membuat hasil akhir optimal sehingga iterasi harus dihentikan; hasil saat ini masih belum optimal sehingga masih diperlukan proses pencarian optimalitas lagi; hasil saat ini adalah hasil optimal sehingga tidak diperlukan iterasi pencarian optimalitas lagi. Pada setiap iterasi pasti akan menghasilkan nilai yang sama atau lebih optimal jika dibandingkan dengan iterasi sebelumnya. Setelah seluruh proses iterasi berhenti, maka hasil akhir dari iterasi tersebut akan digambarkan secara detail ke dalam format laporan yang mudah untuk dimengerti. Penggunaan format laporan berupa visualisasi grafik dan diagram-diagram akan mempermudah user dalam memahami hasil akhir yang dihasilkan oleh tool.

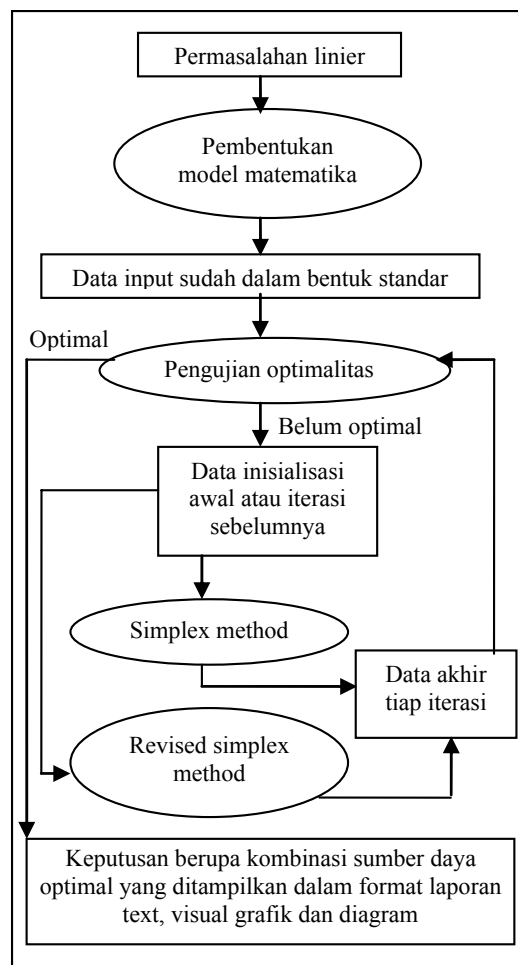
4. PEMBANGUNAN DSS TOOL

Setelah melakukan pemahaman dan perumusan permasalahan linier yang dihadapi oleh user dengan cara mengumpulkan sebanyak mungkin permasalahan dan penyelesaiannya, maka akan didapat gambaran yang jelas mengenai proses yang diperlukan pada DSS tool. Selanjutnya adalah menyusun beberapa alternatif dari data yang telah dikumpulkan, kemudian untuk setiap alternatif yang disusun dibuat pro dan kontra, konsekuensi, serta resiko yang mungkin terjadi, dan yang terakhir memastikan agar semua alternatif yang tersusun haruslah *feasible*.

Kemudian mengevaluasi realistik atau tidaknya alternatif bila dihubungkan dengan tujuan dan sumber daya tersedia. Dan yang terakhir melihat seberapa jauh alternatif tersebut dapat memecahkan masalah. Dilanjutkan dengan pemilihan alternatif terbaik. Keputusan yang dipilih akan diimplementasikan pada permasalahan awal yang ada. Dalam jangka waktu tertentu akan dilakukan evaluasi terhadap efektifitas keputusan yang telah diimplementasikan sebelumnya. Jika keputusan tersebut berjalan dengan baik dan sesuai harapan maka tidak perlu kembali untuk mencari solusi lain.

Sistem yang dibuat memiliki fungsi untuk memberikan kombinasi sumber daya yang tersedia secara optimal sehingga dapat mencapai tujuan minimalisasi atau maksimalisasi yang diinginkan. Input sistem terdiri dari variabel keputusan, fungsi kendala yang membatasi, dan fungsi tujuan yang diinginkan.

Gambar 1 mendeskripsikan alur kerja sistem secara keseluruhan. Pertama kali user akan mendefinisikan permasalahan yang dihadapinya. Selanjutnya user akan mengubah permasalahan tersebut ke dalam model matematika sebelum menginputkannya pada DSS tool ini. Pada tahap ini semua data permasalahan linier sudah siap untuk diproses dengan menggunakan salah satu metode penyelesaian terpilih (simplex atau revised simplex).

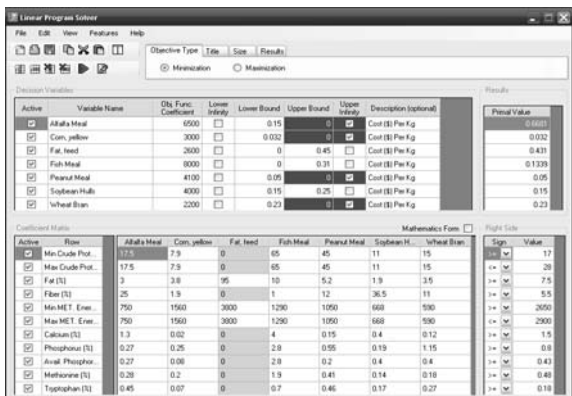


Gambar 1. Arsitektur Sistem

Selanjutnya metode penyelesaian terpilih akan melakukan tahap inisialisasi terlebih dahulu. Tahap inisialisasi ini bertujuan untuk memilah-milah data dan kemudian disimpan pada masing-masing variabel yang telah disiapkan. Tahap selanjutnya yang harus dilakukan adalah melakukan pemeriksaan optimalitas pada permasalahan linier ini. Jika belum optimal maka akan dilakukan proses iterasi guna mencapai hasil yang optimal. Setiap proses iterasi akan selalu menghasilkan nilai yang sama atau lebih optimal daripada iterasi sebelumnya. Ketika proses iterasi berhenti (hasil sudah optimal atau tidak ada solusi untuk permasalahan ini), maka seluruh hasil pada iterasi terakhir yang menggambarkan keadaan optimal permasalahan ini akan digambarkan ke dalam bentuk laporan dengan format yang mudah untuk dimengerti oleh user.

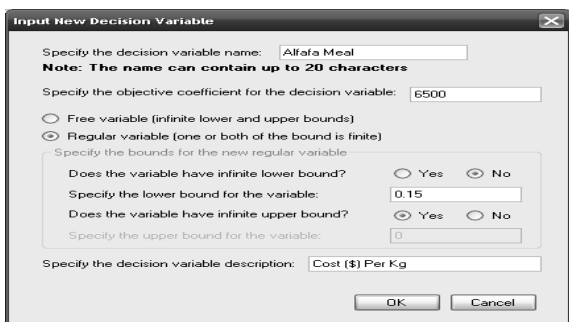
4.1 Fitur-Fitur DSS Tool

Pada sistem yang dikembangkan ini terdapat beberapa fitur utama yang mendukung mulai dari proses penginputan data awal hingga didapatnya hasil optimal pada permasalahan liniernya. Form utama, input variable dan constraint, setting tool, serta pelaporan selama proses iterasi berlangsung dan juga hasil akhir yang diperoleh.



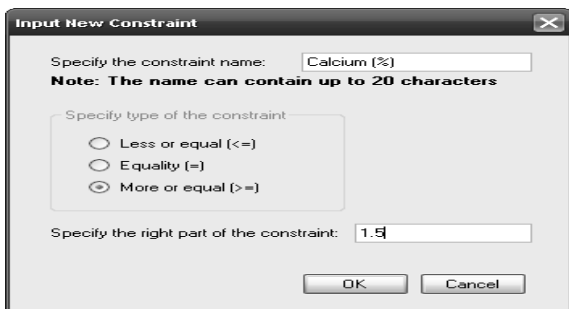
Gambar 2. Form Utama DSS Tool

Pada Gambar 2 menunjukkan tampilan utama dari DSS tool yang dikembangkan. Terdapat beberapa bagian penting seperti grid untuk menampung semua decision variable, constraint, dan juga hasil optimal serta terdapat toolbar untuk setting tool seperti pengaturan metode yang dipilih, pembulatan angka, dan masih banyak lagi.



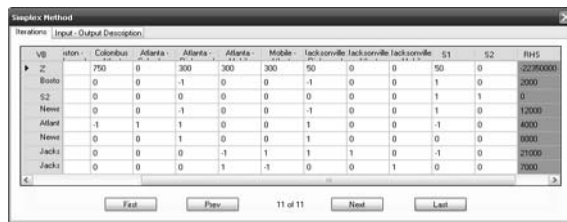
Gambar 3. Form Penginputan Decision Variable

User dapat menambahkan decision variable sebanyak apapun ke dalam tool ini dengan mengisi form pada Gambar 3. Terdapat beberapa field penting yang harus diisi oleh user sebelum mengakhiri pengisian pada form ini.



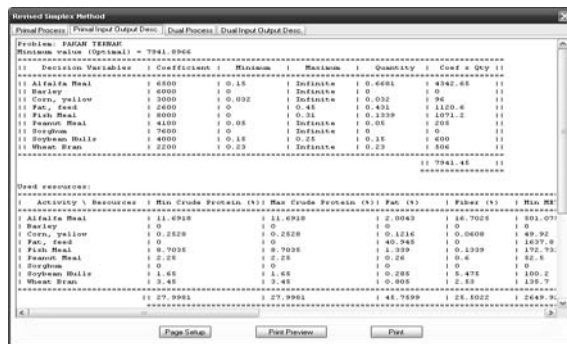
Gambar 4. Form Penginputan Constraint

Untuk menginputkan constraint baru, user dapat memanfaatkan form seperti Gambar 4, dengan menuliskan nama constraint serta tanda dan angka pembatas constraint itu sendiri. Constraint ini berguna sebagai pembatas yang memaksa agar sumber daya dapat mencapai hasil yang optimal.



Gambar 5. Penyelesaian dengan Metode Simplex

Salah satu metode yang ditawarkan untuk menyelesaikan permasalahan adalah simplex dengan memanfaatkan tabel simplex dalam iterasi menuju hasil optimalnya. Pada Gambar 5 ditampilkan iterasi demi iterasi penyelesaian permasalahan dengan metode simplex. User diijinkan untuk dapat melihat iterasi demi iterasi dengan memanfaatkan tombol First, Prev, Next, dan Last.



Gambar 6. Pembacaan Hasil Optimal

Hasil optimal akan dilaporkan dengan salah satu format laporan yaitu mode text seperti yang ditunjukkan pada Gambar 6. Penjelasan detail seperti total biaya dan rincian pengeluaran berguna untuk mempermudah user dalam memahami hasil optimal yang dihasilkan oleh sistem. Sebagai tambahan, user dapat melakukan analisis sensitivitas pada hasil akhir jika dirasa perlu untuk dilakukan.

5. UJICOBANA

Ujicoba dilakukan untuk menguji tingkat fleksibilitas sistem. Hal ini penting untuk dilakukan karena tool ini akan digunakan untuk menyelesaikan permasalahan linier yang bersifat umum (tidak terspesifikasi pada satu bidang permasalahan). Beberapa bidang yang pernah diujicobakan pada tool ini antara lain: kedokteran, produksi, transportasi, penjadwalan, kesehatan.

5.1 Ujicoba Bidang Produksi Pakan Ternak

Pada bidang produksi dapat membantu memberikan solusi alternatif mengenai bagaimana menyusun komposisi pakan ternak yang tepat sehingga dapat meminimumkan biaya produksi tanpa melanggar batas nutrisi pakan ternak yang ideal. Semua batasan telah didefinisikan dengan jelas pada awal penjabaran permasalahan.

Tabel 1. Daftar Bahan Makanan Ternak

<i>Ingredients</i>	<i>Cost</i>	<i>Min</i>	<i>Max</i>
x1 Alfalfa Meal	6.500	0.15	
x2 Corn, yellow	3.000	0.032	
x3 Fat, feed	2.600		0.45
x4 Fish Meal	8.000		0.31
x5 Peanut Meal	4.100	0.05	
x6 Soybean Hulls	4.000	0.15	0.25
x7 Wheat Bran	2.200	0.23	

Terdapat tujuh jenis bahan makanan yang dapat digunakan untuk memproduksi pakan ternak. Pada Tabel 1 dapat diketahui data mengenai masing-masing bahan makanan seperti nama bahan makanan, harga per kg, jumlah minimal dan maksimal yang harus dipenuhi dalam memproduksi pakan ternak. Ketiga faktor tersebut (harga, jumlah minimal, jumlah maksimal) akan berperan penting dalam mengontrol proses optimasi yang akan dilakukan.

Tabel 2. Daftar Kebutuhan Nutrisi Ternak

<i>Nutritions</i>	<i>Min</i>	<i>Max</i>
y1 Crude Protein (%)	17	22
y2 Fat (%)		7.5
y3 Fiber (%)		5.5
y4 MET. Energy (Kcal/kg)	2.650	2.900
y5 Calcium (%)	1.5	
y6 Phosphorus (%)	0.8	
y7 Avail. Phosphorus (%)	0.43	
y8 Methionine (%)	0.48	
y9 Tryptophan (%)	0.18	

Pada Tabel 2 dijelaskan mengenai standar minimum dan maksimum nutrisi, agar ternak dapat tumbuh dengan cepat dan baik. Setiap nutrisi yang akan dihasilkan memiliki batas minimum maupun maksimum yang diperbolehkan. Jumlah minimal dan maksimal tersebut harus dipenuhi agar komposisi makanan menjadi ideal.

Tabel 3. Daftar Kandungan Nutrisi Bahan Makanan

<i>F</i>	<i>x1</i>	<i>x2</i>	<i>x3</i>	<i>x4</i>	<i>x5</i>	<i>x6</i>	<i>x7</i>
y1	17.5	7.9	0	65	45	11	15
y2	3	3.8	95	10	5.2	1.9	3.5
y3	25	1.9	0	1	12	36.5	11
y4	750	1560	3800	1290	1050	668	590
y5	1.3	0.02	0	4	0.15	0.4	0.12
y6	0.27	0.25	0	2.8	0.55	0.19	1.15
y7	0.27	0.08	0	2.8	0.2	0.4	0.4
y8	0.28	0.2	0	1.9	0.41	0.14	0.18
y9	0.45	0.07	0	0.7	0.46	0.17	0.27

Setiap bahan makanan mengandung nutrisi yang berbeda-beda. Perbedaan nutrisi ini akan menjadi salah satu faktor penentu dalam pemilihan kombinasi pakan ternak yang ideal. Daftar kandungan nutrisinya dapat dilihat pada Tabel 3. Semua data ini akan menjadi input dalam DSS tool.

5.1.1 Model Matematika Permasalahan Produksi Pakan Ternak

Untuk mempermudah penyelesaian permasalahan pemilihan komposisi bahan makanan optimal untuk produksi pakan ternak maka permasalahan tersebut diubah dalam model matematika linear programming.

Minimumkan Z:

$$Z = 6500x1 + 3000x2 + 2600x3 + 8000x4 + 4100x5 + 4000x6 + 2200x7$$

Dengan batasan:

$$17.5x1 + 7.9x2 + 65x4 + 45x5 + 11x6 + 15x7 \geq 17$$

$$17.5x1 + 7.9x2 + 65x4 + 45x5 + 11x6 + 15x7 \leq 22$$

$$3x1 + 3.8x2 + 95x3 + 10x4 + 5.2x5 + 1.9x6 + 3.5x7 \geq 7.5$$

$$25x1 + 5x2 + 1.9x2 + x4 + 12x5 + 36.5x6 + 11x7 \geq 5.5$$

$$750x1 + 1560x2 + 3800x3 + 1290x4 + 1050x5 + 668x6 + 590x7 \geq 2650$$

$$750x1 + 1560x2 + 3800x3 + 1290x4 + 1050x5 + 668x6 + 590x7 \leq 2900$$

$$1.3x1 + 0.02x2 + 4x4 + 0.15x5 + 0.4x6 + 0.12x7 \geq 1.5$$

$$0.27x1 + 0.25x2 + 2.8x4 + 0.55x5 + 0.19x6 + 1.15x7 \geq 0.8$$

$$0.27x1 + 0.08x2 + 2.8x4 + 0.2x5 + 0.4x6 + 0.4x7 \geq 0.43$$

$$0.28x1 + 0.2x2 + 1.9x4 + 0.41x5 + 0.14x6 + 0.18x7 \geq 0.48$$

$$0.45x1 + 0.07x2 + 0.7x4 + 0.46x5 + 0.17x6 + 0.27x7 \geq 0.18$$

Dengan batasan tambahan:

$$x1 \geq 0.15 \quad x2 \geq 0.032 \quad x3 \leq 0.45 \quad x4 \leq 0.31$$

$$x5 \geq 0.05 \quad x6 \geq 0.15 \quad x6 \leq 0.25 \quad x7 \leq 0.23$$

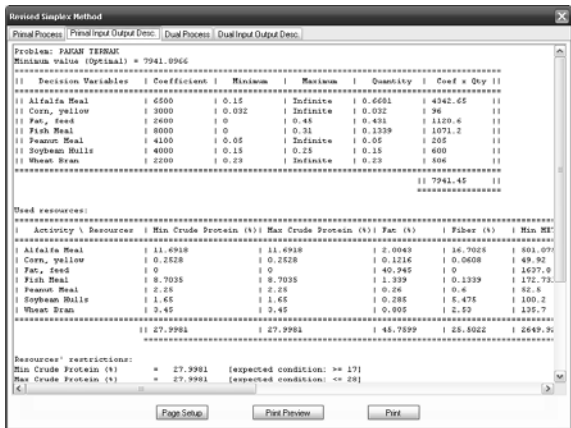
5.1.2 Keputusan dan Kesimpulan Permasalahan Produksi Pakan Ternak

Terbentuknya model matematika permasalahan di atas akan mempermudah dalam pengimplementasian ke dalam DSS tool ini. Setelah menginputkan seluruh decision variables dan constraints serta menentukan objective type permasalahan, yaitu minimalisasi, maka selanjutnya DSS tool akan membantu memberikan solusinya. Menurut DSS tool, untuk solusi permasalahan pertama akan didapat hasil kombinasi pakan ternak terbaik yaitu sebagai berikut:

Tabel 4. Kombinasi Optimal Pakan Ternak

<i>Ingredients</i>	<i>Qty</i>	<i>Total Cost</i>
Alfalfa Meal	0.6681	4.342,65
Corn, yellow	0.032	96
Fat, feed	0.431	1.120,6
Fish Meal	0.1339	1.071,2
Peanut Meal	0.05	205
Soybean Hulls	0.15	600
Wheat Bran	2.200	506
Total Biaya Produksi		7.941,45

Menurut DSS tool untuk kombinasi optimal bahan makanan pakan ternak adalah seperti terlihat pada Tabel 4. Dituliskan jumlah bahan makanan yang diperlukan dan total biayanya. Dengan memanfaatkan DSS tool, permasalahan optimasi produksi pakan ternak dapat diselesaikan dengan cepat.



Gambar 7. Pelaporan Rincian Hasil Optimal

Untuk melengkapi pelaporan tool terhadap user, maka diberikan rincian penggunaan sumber daya yang tersedia seperti yang ditunjukkan pada Gambar 7. Hal ini tentunya akan lebih memudahkan user dalam mengambil keputusan.

5.2 Ujicoba Transportasi Pengiriman Barang

Wilayah Newark (20.000 unit) dan Jacksonville (28.000) mempunyai tugas untuk supply mobil ke lima wilayah lainnya Boston (10.000), Columbus (6.000), Richmond (10.000), Atlanta (17.000), Mobile (7.000). (John S Loucks, 2003).

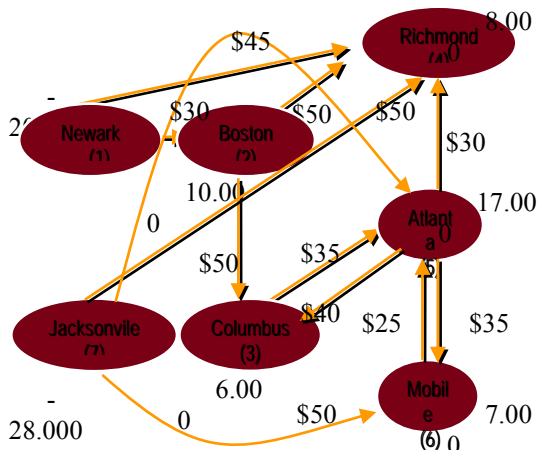
Tabel 5. Biaya Pengiriman Antar Wilayah

From	To	Cost (\$) per unit
1 Newark	2 Boston	300
1 Newark	4 Richmond	400
2 Boston	3 Columbus	500
2 Boston	4 Richmond	500
3 Columbus	5 Atlanta	350
5 Atlanta	3 Columbus	400
5 Atlanta	4 Richmond	300
5 Atlanta	6 Mobile	350
6 Mobile	5 Atlanta	250
7 Jacksonville	4 Richmond	500
7 Jacksonville	5 Atlanta	450
7 Jacksonville	6 Mobile	500

Tabel 5 menunjukkan biaya pengiriman per unit mobil dari satu wilayah ke wilayah lainnya yang mampu dijangkau oleh wilayah pengirim tersebut.

5.2.1 Model Matematika Permasalahan Pemilihan Jalur Pengiriman Barang

Permasalahan yang ada pada akan diubah ke dalam bentuk model matematika sebelum diimplementasikan pada DSS tool. Untuk memperjelasnya dapat dibuat terlebih dahulu state pengiriman antar wilayahnya. Gambar 8 mendeskripsikan alur pengiriman barang dari satu wilayah ke wilayah lainnya beserta detail supply dan demand per wilayah, serta biaya pengiriman per unit barangnya.



Gambar 8. State Pengiriman Antar Wilayah

Minimumkan Z:

$$Z = 300x_{12} + 400x_{14} + 500x_{23} + 500x_{24} + 350x_{35} + 400x_{53} + 300x_{54} + 350x_{56} + 250x_{65} + 500x_{74} + 450x_{75} + 50x_{76}$$

Dengan kendala:

$$\begin{aligned} -x_{12} - x_{14} &\geq -20000 \\ x_{12} - x_{23} &= 10000 \\ x_{23} + x_{53} - x_{35} &= 6000 \\ x_{14} + x_{54} + x_{74} &= 8000 \\ x_{35} + x_{65} + x_{75} - x_{53} - x_{54} - x_{56} &= 17000 \\ x_{56} + x_{76} - x_{65} &= 7000 \\ -x_{74} - x_{75} - x_{76} &\geq -28000 \end{aligned}$$

5.2.2 Keputusan dan Kesimpulan Permasalahan Pemilihan Jalur Pengiriman Barang

Setelah membuat permodelan bentuk matematika dari sebuah proses pendistribusian pengiriman barang dari satu daerah ke daerah lain dan kemudian menerapkannya ke dalam tool ini, maka tool ini akan menunjukkan bagaimana alur pengiriman dan jumlah barang yang dikirim ke setiap daerah dengan tepat sehingga biaya operasional dapat diminimalkan.

The screenshot shows a 'Decision Variables' table with columns for Active, Variable Name, Obj. Func. Coefficient, Lower Bound, Upper Bound, and Description (optional). The 'Results' column shows the optimal values for each variable.

Active	Variable Name	Obj. Func. Coefficient	Lower Bound	Upper Bound	Description (optional)	Results
<input checked="" type="checkbox"/>	Newark-Boston	300	0	0	Newark - Boston	0
<input checked="" type="checkbox"/>	Newark-Richmond	400	0	0	Newark - Richmond	8000
<input checked="" type="checkbox"/>	Boston-Columbus	500	0	0	Boston - Columbus	2000
<input checked="" type="checkbox"/>	Boston-Richmond	500	0	0	Boston - Richmond	0
<input checked="" type="checkbox"/>	Columbus-Atlanta	350	0	0	Columbus - Atlanta	0
<input checked="" type="checkbox"/>	Jacksonville-Richmond	500	0	0	Jacksonville - Richmond	0
<input checked="" type="checkbox"/>	Jacksonville-Atlanta	450	0	0	Jacksonville - Atlanta	21000
<input checked="" type="checkbox"/>	Jacksonville-Mobile	500	0	0	Jacksonville - Mobile	7000
<input checked="" type="checkbox"/>	Mobile-Atlanta	250	0	0	Mobile - Atlanta	0
<input checked="" type="checkbox"/>	Atlanta-Columbus	400	0	0	Atlanta - Columbus	4000

Gambar 9. Hasil Optimal Pemilihan Jalur

Sistem yang dikembangkan ini mampu dengan baik memberikan solusi pemilihan jalur pengiriman dengan pemakaian biaya terendah (optimal). Pada Gambar 9 terlihat hasil akhir optimal pemilihan wilayah pengiriman barang antar wilayah beserta jumlah barang yang dikirimkan dari satu wilayah ke wilayah lainnya. DSS tool yang dikembangkan ini dapat memberikan alternatif solusi penyelesaian dengan cepat dan tepat.

Tabel 6. Jalur Pengiriman Barang

From		To		Qty	Cost
1	Newark	2	Boston	12.000	3.600.000
1	Newark	4	Richmond	8.000	3.200.000
2	Boston	3	Columbus	2.000	1.000.000
5	Atlanta	3	Columbus	4.000	1.600.000
7	Jacksonville	4	Richmond	21.000	9.450.000
7	Jacksonville	5	Atlanta	7.000	3.500.000
Total Cost					22.350.000

Tabel 6 menggambarkan alur pengiriman terbaik antar wilayah beserta jumlah unit yang harus dikirimkan. Total biaya yang dihasilkan merupakan total biaya minimum (optimal).

Pada ujicoba yang dilakukan terhadap kedua bidang di atas, tidak ditemukan kendala-kendala yang mengganggu selama proses penginputan data hingga didapatnya hasil akhir optimal pada masing-masing permasalahan. Dibutuhkan waktu yang sangat singkat untuk menghasilkan hasil yang optimal dengan menggunakan DSS tool ini.

Selain keenam bidang di atas, tool ini dapat juga digunakan pada bidang-bidang lain yang berhubungan dengan pencarian optimalitas selama masih menggunakan fungsi-fungsi yang sifatnya linier.

6. PENUTUP

Metode linear programming dapat digunakan untuk menghasilkan suatu keputusan yang optimal pada permasalahan optimalisasi sumber daya yang terbatas. Linear programming sangat baik untuk menyelesaikan permasalahan yang sifatnya linear.

Penggunaan simplex method maupun revised simplex method akan memberikan hasil yang sama. Namun setiap metode tersebut memiliki keunggulan masing-masing. Pada simplex method, prosesnya lebih mudah untuk dimengerti karena digambarkan dengan jelas pada tabel simplex, sedangkan pada revised simplex method, proses iterasi yang dibutuhkan untuk menyelesaikan permasalahan yang berskala besar membutuhkan jumlah iterasi yang lebih sedikit dari penggunaan simplex method.

Penggunaan DSS tool ini dapat membantu menghemat waktu yang diperlukan user untuk mengambil keputusan optimal dengan cepat dan tepat daripada menggunakan perhitungan secara manual.

Terkadang tidak mudah untuk mengimplementasikan permasalahan pada dunia nyata untuk diterapkan pada tool ini. Diperlukan sedikit kemampuan matematika sehingga user bisa mengubah permasalahan tersebut menjadi bentuk model matematika sebelum DSS tool ini dapat memprosesnya lebih lanjut guna menghasilkan sebuah keputusan yang berguna bagi user.

DSS tool ini berguna untuk memberikan alternatif solusi kepada user tanpa bermaksud untuk memaksakan atau menggantikan peranan dari user itu sendiri dalam hal penentuan pengambilan keputusan.

PUSTAKA

- Hermawan, Julius. *Membangun Decision Support System*. Andi. 2005.
- Kakiay, Thomas. *Pemrograman Linear: Metode & Problema*. Andi. 2008.
- Lieberman, Hillier. *Introduction To Operations Research (Eighth Edition)*. Andi. 2008.
- Siringoringo, Hotniar. *Seri Teknik Riset Operasional Pemrograman Linear*. Graha Ilmu. 2005.
- Turban, Erfaim. *Decision Support System and Intelligent System*. Andi. 2005.
- Wagner, Harvey M. *Priciples of Operation Research*. 2003.
- Winston, Wayne L. *Operation Research: Aplications and Algorithm*. Duxbury Press, California. 2003.