

PENGEPAKAN PIPA BERAGAM DIAMETER KE DALAM SATU KONTAINER DENGAN MENGGUNAKAN SOLUSI HEURISTIK ALGORITMA GENETIKA

Moses L. Singgih

Dosen Fakultas Teknologi Industri, Jurusan Teknik Industri - Institut Teknologi Sepuluh Nopember

Sundoro Untung

Dosen Fakultas Teknologi Industri, Jurusan Teknik Industri - Institut Teknologi Sepuluh Nopember

ABSTRAK

Literatur mengenai permasalahan pengepakan (*packing problem*) secara umum telah dipelajari secara intensif dan sangat luas pengembangannya. Salah satu masalah pengepakan yang ada adalah masalah pengepakan silinder. Pada tulisan ini diteliti mengenai masalah yang merupakan bagian dari pengepakan silinder yaitu pengepakan pipa. Dalam pengepakan pipa digunakan model matematis non linier mixed integer programming untuk menerangkan permasalahan pengepakan secara matematis dan menyarankan menggunakan metode heuristik algoritma genetika (*genetic algorithm*) dipadukan dengan penggunaan daftar iterasi tabu untuk menyelesaikan model matematis yang dihadapi sehingga solusi yang dihasilkan dapat memuaskan ditinjau dari sisi konfigurasi pengepakan dan juga efisien ditinjau dari waktu komputasi. Algoritma yang diusulkan pada tulisan ini mampu memberikan konfigurasi pengepakan pipa yang memuaskan bila ditinjau dari pemanfaatan ruangan dan waktu komputasi yang cepat.

Kata kunci: pengepakan pipa, model matematis, Algoritma Genetika.

ABSTRACT

Literature on packing problems in general is extensive and too broad to be covered here. One of the packing problems is packing for cylindrical material, that is packing for pipe. To pack the pipes, a mathematical model non linear mixed integer programming will be applied to explain packing problem mathematically. A combined genetic algorithm and tabu list is suggested to solve the mathematical problem to obtain satisfactory solution in terms of packing configuration and computational time. The algorithm gives a good result in terms of space utilization and computational time.

Keywords: pipe packing, mathematical model, genetic algorithm.

1. PENDAHULUAN

Literatur mengenai permasalahan pengepakan (*packing problem*) secara umum telah dipelajari secara intensif dan sangat luas pengembangannya. Beberapa penulis yang telah membahas permasalahan ini antara lain Haessler dan Sweeney (1991), Dowsland dan Dowsland (1990, 1992) maupun Chen, et. al. (1995). Haessler, Sweeney, dan Dowsland membahas pengepakan kotak, baik dua maupun tiga dimensi. Sedangkan Chen, et. al. membahas model analitis untuk mengoptimalkan permasalahan pengepakan kotak ke dalam kotak secara tiga dimensi.

Penelitian mengenai pengepakan juga semakin berkembang, beberapa pengarang telah pula melakukan penelitian mengenai pengepakan silinder dengan 'ukuran yang sama' ke dalam satu kotak, diantaranya Dowsland (1991), Fraser dan George (1994) maupun

Isermann (1991). Isermann memberikan solusi heuristik untuk pengepakan silinder homogen secara umum, Fraser dan George mendiskusikan mengenai pengepakan silinder dengan ukuran yang sama dalam kontainer dengan dimensi yang telah diketahui dan menerapkannya untuk menyelesaikan permasalahan pengepakan kertas gulung (roll kertas). Dowsland menjelaskan bagaimana menentukan dimensi optimal bahan untuk pengepakan produk-produk silinder.

Pentingnya penelitian mengenai pengepakan silinder didasari atas pertimbangan bahwa kesalahan dalam penempatan silinder akan menyebabkan adanya ruang yang terbuang dalam kontainer dan dapat menyebabkan pemborosan dari sisi biaya.

Suatu pengembangan yang lebih bisa diterapkan untuk pengepakan silinder dilakukan oleh George, J.A., et. al. (1995) yang mengembangkan model dan metode solusi secara heuristik untuk permasalahan pengepakan silinder dengan berbagai diameter ke dalam satu kotak tertentu. Mereka menggunakan model matematis *non linier mixed integer programming* untuk menerangkan permasalahan pengepakan secara matematis dan menyarankan menggunakan metode heuristik algoritma genetika (*genetic algorithm*) untuk menyelesaikan model matematis yang dihadapi.

Dalam tulisan ini dikembangkan metode yang dibahas oleh George, et. al. (1995). Metode yang dibahas oleh George, et. al. (1995) mengabaikan unsur memasukkan silinder ke dalam silinder (*nested pipes*) ditambah karena penelitian yang ada selama ini adalah menggunakan silinder pejal, walaupun sebenarnya unsur ini telah diidentifikasi sebagai 1 dari 3 masalah yang dapat timbul pada pengepakan silinder oleh John A. George. Ketiga permasalahan tersebut secara lengkap yaitu :

- Bagaimana mengoptimalkan pengepakan untuk kasus silinder berongga, dimana bisa dimungkinkan terjadi persarangan satu silinder ke dalam silinder lain.
- Bagaimana mengoptimalkan pengepakan silinder pejal ke dalam satu kontainer/kotak, optimal disini meliputi jenis apa saja yang harus dimasukkan dan bagaimana pula penataannya.
- Bagaimana mengalokasikan silinder ke dalam berbagai kontainer/kotak dengan berbagai ukuran untuk meminimasi jumlah kontainer yang akan dikirimkan.

Gabungan masalah pertama dan kedua inilah yang diteliti dalam tulisan ini. Dengan memperhatikan unsur memasukkan silinder ke dalam silinder ini maka teori pengepakan dari George, et. al. (1995) yang hanya memecahkan masalah kedua akan lebih berkembang dan tentunya akan dapat lebih luas penerapannya. Dalam membahas masalah tersebut, dalam tulisan ini diasumsikan bahwa pembahasan dilakukan dengan mengabaikan faktor terjadinya kesulitan untuk mengurutkan silinder, memasukkan silinder ke dalam silinder ataupun menarik kembali silinder yang telah ditata untuk kemudian digunakan, panjang silinder dianggap sama dengan salah satu dimensi dari kontainer sehingga permasalahan dibatasi pada permasalahan pengoptimalan silinder pada persegi panjang (2 dimensi) dan tidak terjadi perubahan bentuk pada silinder bila diberikan beban di atasnya, beban disini berupa silinder lain.

2. FORMULASI MODEL MATEMATIS PENGEPAKAN PIPA

Dalam bagian ini, pengepakan pipa dengan berbagai diameter ke dalam satu kontainer diformulasikan sebagai *non linier mixed integer programming*. Pengepakan silinder pejal

telah dipublikasikan (George, et. al., 1995) namun permasalahan pengepakan pipa ini memiliki struktur yang lebih rumit.

2.1 Data dan Variabel Keputusan

Set indeks I menunjukkan kumpulan dari pipa sebagai kandidat yang akan ditempatkan ke dalam kotak. Rl_i adalah jari-jari luar dari pipa ke- i . Rd_i adalah jari-jari dalam dari pipa ke- i . Walaupun ada beberapa pipa yang berdiameter sama, nilai sebuah i tetap hanya untuk 1 pipa. A dan B masing-masing menunjukkan dimensi horisontal dan dimensi vertikal dari kontainer. X_i dan Y_i menunjukkan koordinat dari tiap pipa. d_{ij} bernilai 1 apabila pipa ke- i diletakkan di dalam pipa ke- j dengan perkecualian bahwa indeks j bernilai 0 apabila pipa ke- i langsung bersentuhan dengan kontainer. W_i menunjukkan 'keuntungan' apabila pipa ke- i dimasukkan ke dalam kotak. Ada beberapa alternatif untuk menentukan nilai W (bobot untuk pipa), antara lain :

- Densitas silinder yang termuat dalam kontainer.
- Harga jual dari silinder yang termuat dalam kontainer.
- Ruang kosong yang tersisa dalam kontainer.
- Keuntungan finansial karena silinder dapat dimuat dalam kontainer.
- Prioritas silinder

Dalam tulisan ini, digunakan densitas (kepadatan) silinder yang termuat dalam kontainer. Baik nilai jual maupun keuntungan mempunyai kelemahan yaitu sering berfluktuasi seiring dengan waktu. Sedangkan ruang kosong yang tersisa akan sulit dihitung terutama karena adanya penempatan silinder ke dalam silinder. Prioritas silinderpun sering berubah-ubah dan biasanya tergantung kepada kondisi penggunaannya.

2.2 Formulasi Matematis

Fungsi tujuan memaksimalkan $\sum_{i \in I} W_i d_i$

Fungsi-fungsi kendala:

- (a) $d_i Rl_i \leq X_i \leq d_i (A - Rl_i)$ " $i \in I$
- (b) $d_i Rl_i \leq Y_i \leq d_i (B - Rl_i)$ " $i \in I$ dimana $\delta_i = \sum_{k \in I} d_{ik}$
- (c) $(d_{ik} X_i - d_{jk} X_j)^2 + (d_{ik} Y_i - d_{jk} Y_j)^2 \leq d_{ik} d_{jk} (Rl_i + Rl_j)^2$ " $i, j, k \in I, i > k, j > k, i \neq j$
- (d) $(d_{ik} d_{kj} X_i - d_{kj} d_{ik} X_k)^2 + (d_{ik} d_{kj} Y_i - d_{kj} d_{ik} Y_k)^2 \leq d_{ik} d_{kj} (Rl_i - Rd_k)^2$
" $i, j, k \in I, i > k, k > j, i \neq k, k \neq j$
- (e) $\sum_{k \in I} d_{ik} \leq 1$ " $i \in I$
- (f) $X_i \geq 0, Y_i \geq 0$ " $i \in I$
- (g) $d_{ij} \in \{0, 1\}$ " $i, j \in I$

Kendala (a) dan (b) menjamin jika pipa telah dimasukkan ke dalam kotak, maka tidak ada bagian dari pipa yang keluar dari kotak. Kendala (c) menjamin tidak ada satu pipa yang berpotongan dengan pipa lain dalam lingkup pipa yang sama, hubungan antar pipa dalam lingkup pipa yang sama minimal adalah bersinggungan. Kendala (d) menjamin pipa yang berada di dalam tidak berpotongan dengan pipa yang berada diluarnya. Kendala (e) menjamin satu pipa hanya ditempatkan dalam satu pipa saja, yaitu pipa lain atau bersentuhan langsung dengan kotak. Kendala (f) bersifat sebagai penentu koordinat sumbu x dan y untuk pipa ke- i . Kendala (g) sebagai variabel binari yang bernilai 1 bila

pipa ke- i ditempatkan ke dalam kotak atau pipa lain (pipa ke- j) dan bernilai 0 bila tidak ditempatkan ke dalam kotak. Penggunaan metode heuristik untuk memecahkan permasalahan diatas dilakukan dengan alasan :

- Formulasi diatas mempunyai struktur yang sulit, yaitu merupakan gabungan variabel *integer* dan variabel *non integer* serta merupakan program *non linear*.
- Model matematis dari permasalahan diatas mempunyai variabel yang berkembang dalam jumlah, secara cepat seiring dengan meningkatnya jumlah silinder yang akan dimasukkan.

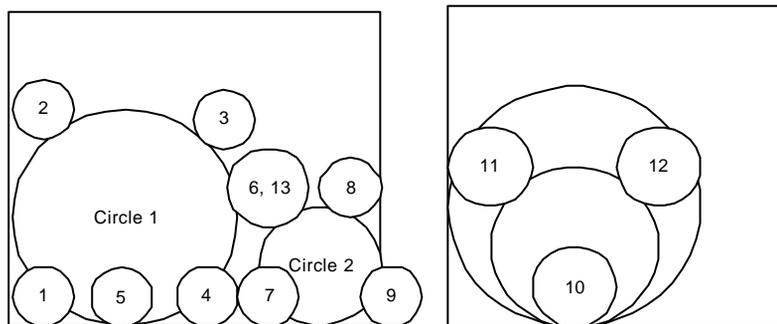
3. METODE SOLUSI

Bagian ini memberikan penjelasan mengenai komponen-komponen yang diperlukan untuk membuat satu algoritma solusi heuristik.

3.1 Angka Posisi

Dalam tulisan ini, tidak lagi menggunakan angka sisi dari penelitian sebelumnya, karena mempunyai kelemahan. Kelemahan dari angka sisi adalah hanya menjelaskan adanya 3 sisi tempat yang mungkin dari satu lingkaran untuk meletakkan lingkaran berikutnya. Ketiga tempat tersebut adalah sisi kiri, sisi kanan dan sisi bawah. Menurut logika, bila ada satu lingkaran yang dikelilingi oleh 3 sisi (kiri, kanan dan bawah), maka setidaknya ada 4 tempat yang mungkin bagi lingkaran berikutnya untuk ditempatkan, yaitu sisi kiri atas, sisi kiri bawah, sisi kanan atas dan sisi kanan bawah. Adanya kelemahan tersebut ditambah dengan munculnya tempat-tempat baru akibat lingkaran yang berongga, mendorong penulis mengembangkan model angka posisi menjadi sebagai berikut.

Untuk tiap lingkaran k , misalkan f_k adalah angka posisi yang mungkin, dimana f_k bernilai 2 (sudut kiri bawah dan sudut kanan bawah) untuk $k = 1$ dan bernilai $(k^2 + 11k - 16) / 2$ untuk $k \geq 2$.



Gambar 1. Ada 13 Posisi Kemungkinan Penempatan Lingkaran Ketiga Bila Ada 2 Lingkaran yang Telah Ditempatkan

Berikut adalah tabel kemungkinan berbagai posisi yang diwujudkan dalam angka sisi dan posisi untuk lingkaran k bila lingkaran i dan j telah ditempatkan.

Tabel 1. Angka Posisi Sebagai Alat Bantu Pengepakan Pipa

Ling- karan ke-	Kiri Atas	Kiri Bawah	Kanan Atas	Kanan Bawah	Menyen- tuh Dasar	Kiri Dalam	Kana n Dala m	Lingkaran lain ke-		
								1	2	..
1	1	2	3	4	5					
2	6	7	8	9	10	11	12	13		
3	14	15	16	17	18	19	20	21	22	
..

Sesuai dengan Gambar 1 maka ada 13 posisi yang mungkin apabila ada dua lingkaran yang telah ditempatkan di dalam kotak.

3.2 Array Posisi

Untuk mengevaluasi setiap posisi yang mungkin untuk mengalokasikan tiap lingkaran dalam kotak, maka perlu terlebih dahulu mengalokasikan lingkaran pada *initial* posisi. Informasi bentuk ini disimpan dalam *array* posisi, yang berisi angka posisi. Dengan mengkombinasikan informasi dari *array* yang berbeda, konsep angka posisi sesuai dengan kerangka algoritma genetika.

Misalkan P adalah *array* yang berisi daftar dari angka posisi untuk semua $k \in I$, sesuai urutan diameter dan p_k adalah angka posisi untuk lingkaran ke- k . Untuk mengilustrasikan penggunaan angka sisi dan angka posisi serta *array* posisi akan ditunjukkan dalam contoh berikut.

Sebuah *array* yang mungkin dapat terjadi adalah 1, 4, 14, 8, 3, ... , 23. Untuk lingkaran yang akan ditempatkan ke dalam kotak, tidak semua angka posisi yang keluar adalah mungkin untuknya. Untuk lingkaran pertama hanya angka posisi 1 yang mungkin. Untuk lingkaran ketiga hanya angka posisi 1 sampai 13 yang mungkin, walaupun ada yang tidak layak untuk ditempati dari sisi ukuran ruangnya (terlalu kecil). Begitu sebuah *array* telah dibangkitkan, harus dilakukan penguraian kode (*decode*). Misalkan untuk lingkaran $k = 2$, maka $p_k = 4$ sesuai *array* yang ada, dan $f_k = 13$, maka prosedur pencarian lokasi yang layak dimulai dari angka posisi 4, 5, ... , 13 lalu 1, 2, ... , 3. Jika tidak ada angka posisi yang layak untuk ditempati, maka lingkaran ke-2 diabaikan dan dilanjutkan lingkaran ke-3.

3.3 Algoritma Genetika

Algoritma Genetika mempunyai apa yang disebut sebagai individu, tiap individu ini mempunyai karakteristik yang ditandai oleh nilai ketepatan (*fitness*). Nilai ketepatan ini bersesuaian dengan nilai fungsi tujuan yang dihasilkan oleh penggunaan individu tadi. Algoritma Genetika bekerja secara iteratif, dimana tiap iterasi dinamakan generasi. Sebuah generasi mempunyai serangkaian individu, tiap individu dihitung nilai ketepatannya. Dalam pembentukan generasi berikutnya individu yang memiliki nilai ketepatan tertinggi dikawinkan atau dimutasikan untuk menghasilkan anak bagi generasi berikutnya. Perkawinan dilakukan dengan menggabungkan *array* (*gen*) dari 2 individu terpilih. Sedangkan mutasi dilakukan pada *array* (*gen*) dari 1 individu terbaik. Untuk

mempertahankan populasi dalam generasi tersebut maka individu dengan nilai ketepatan rendah dapat dimatikan. Keefektifan dari pendekatan ini bergantung pada seberapa baik karakteristik dari satu generasi dari *array* yang berlanjut ke generasi berikutnya.

Langkah-langkah yang harus ditempuh dalam menerapkan pendekatan Algoritma Genetika yang disarankan oleh George, et. al. (1995) ini adalah :

- a. Serangkaian *array* posisi dibangkitkan dengan menggunakan proses randomisasi.
- b. Pada tiap-tiap *array* posisi ini dilakukan 'dekodisasi' yang berguna untuk menjabarkan angka random dari proses randomisasi ke permulaan angka posisi. Dengan menggunakan *array* posisi yang ada, ditentukan konfigurasi pengepakan untuk tiap *array* posisi dan dihitung nilai fungsi tujuan masing-masing *array* posisi sebagai pengukur ketepatan tiap *array*.
- c. *Array* yang terpilih diproduksi untuk generasi berikutnya. Probabilitas sebuah *array* diproduksi proposional dengan rasio ketepatan *array* tersebut terhadap jumlah ketepatan seluruh *array* yang dibangkitkan.
- d. Pertukaran pasangan dari tiap *array* dari tiap populasi yang direproduksi ditentukan dengan menggunakan sebuah angka random sebagai penanda letak pertukaran. Sebagai contoh *Parent 1* mempunyai *array* posisi 1, 5, 4, 2, 7, 3, 6 dan *Parent 2* mempunyai *array* posisi 7, 2, 4, 6, 3, 5, 1. Titik pertukaran terpilih secara random adalah titik ke-3. Maka 2 *parent* tersebut akan menghasilkan 2 anak yaitu anak 1 dengan *array* posisi 1, 5, 4, **6, 3, 5, 1** dan anak 2 dengan *array* posisi 7, 2, 4, **2, 7, 3, 6**.
- e. Algoritma tersebut dihentikan bila semua *array* posisi memiliki nilai fungsi tujuan yang sama atau berdekatan dengan toleransi tertentu.

Langkah-langkah ini masih relevan untuk digunakan pada permasalahan pengepakan pipa, hanya muncul satu masalah mengenai kelambanan komputasi.

3.4 Penggunaan Daftar Iterasi Tabu

Algoritma *Neighborhood Search* mempunyai 2 unsur penting yang biasanya saling bertentangan yaitu lama iterasi dan ketepatan dari algoritma yang digunakan. Algoritma Genetika sudah dibuktikan tepat untuk digunakan dalam permasalahan pengepakan silinder oleh John A. George. Namun bila dilihat dari lama iterasi, Algoritma Genetika saja tidak cukup efisien. Untuk jumlah silinder pejal antara 6 hingga 40, rata-rata tiap permasalahan iterasi komputasinya memerlukan waktu sampai hampir 5 menit.

Untuk permasalahan silinder berongga, dengan jumlah silinder sekitar 250 silinder, Algoritma Genetika akan memakan waktu yang lama, apalagi iterasi ke- i akan lebih lama daripada iterasi ke- $(i-1)$. Sebagai solusinya, disarankan untuk menggabungkan Algoritma Genetika dengan prinsip yang mengadopsi Algoritma *neighborhood search* yang lain yaitu *Tabu Search*. Prinsip *Tabu Search* adalah membuat daftar yang berisi iterasi yang tabu untuk dilakukan. Ide dari penggunaan Daftar Tabu disini adalah bahwa bila ukuran diameter silinder ke- i sama dengan silinder ke- $(i-1)$, dan ternyata silinder ke- i tersebut tidak bisa ditempatkan pada angka posisi tertentu maka pastilah silinder ke- $(i-1)$ tidak bisa ditempatkan, kemudian angka posisi ini dimasukkan ke dalam Daftar Tabu. Bila suatu angka posisi terdapat dalam Daftar Tabu iterasi tidak perlu dilakukan. Sebagai gambaran betapa banyaknya waktu yang dapat dihemat akibat iterasi yang tak perlu, dapat diilustrasikan sebagai berikut. Bila silinder yang ditempatkan adalah silinder ke- i dan angka posisi yang muncul adalah x , maka diperlukan proses pengujian apakah

penempatan silinder ke- i di x akan berpotongan dengan silinder 1 hingga silinder $i - 1$. Bila berpotongan angka posisi diubah kemudian dilakukan proses pengujian kembali. Dengan adanya Daftar Tabu, maka angka posisi yang masuk ke dalam Daftar Tabu tidak perlu diuji.

4. HASIL KOMPUTASI

4.1 Data Eksperimen

Untuk kepentingan validasi algoritma, digunakan ukuran kontainer dengan panjang 6 meter, lebar 1,5 m dan tinggi 2 meter. Data mempunyai fungsi untuk memvalidasi algoritma yang dibuat. Untuk itu dari berbagai macam jenis silinder yang telah dicantumkan harus dilakukan pemilihan agar proses validasi data dapat terarah dan sesuai dengan tujuannya.

Dalam tulisan ini, digunakan 5 jenis diameter luar dari silinder berongga yaitu diameter 90 mm ketebalan 5.4 mm (jenis kecil 1), diameter 110 mm ketebalan 6.6 mm (jenis kecil 2), diameter 280 mm ketebalan 16.6 mm (jenis sedang 1), diameter 315 mm ketebalan 18.7 mm (jenis sedang 2) dan diameter 500 mm ketebalan 29.6 mm (jenis besar).

Berdasarkan percobaan sederhana pada rancangan *software*, diameter jenis kecil 1 saja akan memenuhi kontainer dalam jumlah yang kurang dari 350 silinder, diameter jenis kecil 2 saja akan memenuhi kontainer dalam jumlah yang kurang dari 265 silinder, diameter jenis sedang 1 saja akan memenuhi kontainer dalam jumlah yang kurang dari 40 silinder, diameter jenis sedang 2 saja akan memenuhi kontainer dalam jumlah yang kurang dari 30 silinder sedangkan diameter jenis besar saja akan memenuhi kontainer dalam jumlah yang kurang dari 12 silinder.

Yang perlu dicermati disini adalah keefektifan peletakan silinder di dalam silinder dan peletakan antar silinder dalam kontainer. Percobaan yang dilakukan disini adalah dengan memodifikasi jumlah yang akan ditempatkan untuk masing-masing jenis. Jenis kecil 1 dan 2 akan dipakai secara bergantian dalam artian tidak dipakai bersama-sama, demikian pula dengan jenis sedang 1 dan sedang 2. Secara lengkap skema percobaan tampak dalam Tabel 2.

4.2 Hasil Eksperimen

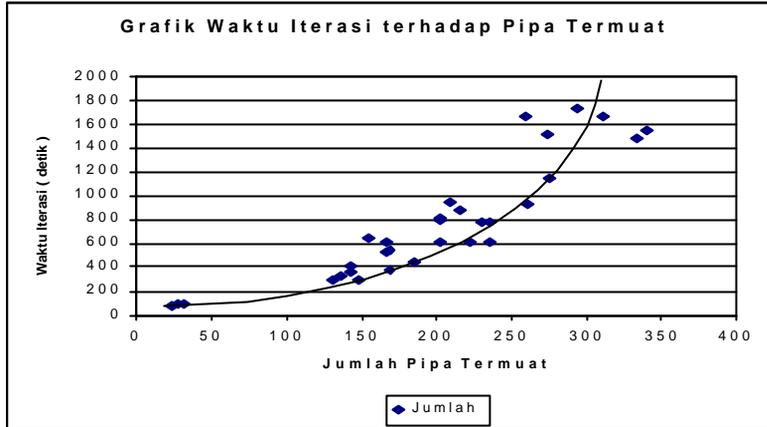
Dari serangkaian hasil eksperimen yang dilakukan, didapatkan bahwa secara umum Algoritma Genetika akan memberikan nilai fungsi tujuan yang lebih baik. Sehingga dapat dikatakan bahwa Algoritma Genetika seringkali dapat meningkatkan nilai fungsi tujuan dan memberikan jumlah silinder yang termasuk dalam kontainer lebih banyak. Pada beberapa eksperimen yang dilakukan memang ada kejadian dimana nilai fungsi tujuan tidak bertambah baik walaupun telah dilakukan proses iterasi. Ada 2 penyebab dari kejadian ini, yang pertama adalah konfigurasi individu awal memang telah optimal, dan penyebab yang kedua adalah 'ketidakteruntungan' angka acak untuk mengawinkan individu-individu sehingga apa yang dihasilkan oleh individu baru tidak lebih baik dari individu lama.

Tabel 2. Skema Lengkap Eksperimen yang Dilakukan

Eksperimen	Jumlah silinder untuk diameter				
	90 mm	110 mm	280 mm	315 mm	500 mm
1	350				5
2		265			5
3	350				12
4		265			12
5			40		5
6				30	5
7			40		12
8				30	12
9	350			10	
10		265		10	
11	350		20		
12		265	20		
13	350			30	
14		265		30	
15	350		40		
16		265	40		
17		265		10	5
18		265	20		5
19		265		30	5
20		265	40		5
21		265		10	12
22		265	20		12
23		265		30	12
24		265	40		12
25	350			10	5
26	350		20		5
27	350			30	5
28	350		40		5
29	350			10	12
30	350		20		12
31	350			30	12
32	350		40		12

Bila ditinjau dari ketepatan penggunaan angka posisi, dapat dikatakan penggunaannya sudah tepat. Semua tempat tersisa memang sudah tidak dapat lagi dimasukkan silinder kedalamnya.

Ditinjau dari segi waktu, satu iterasi memakan waktu antara 1 hingga 15 detik tergantung kepada jumlah silinder yang dimasukkan dan skema konfigurasi yang dihasilkan. Secara keseluruhan grafik perbandingan antara waktu iterasi dengan jumlah silinder termuat tampak dalam Gambar 2 berikut.



Gambar 2. Grafik Perbandingan Waktu Iterasi terhadap Pipa Termuat

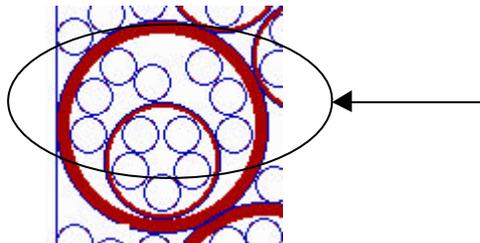
Dari Gambar 2 diatas dapat dilihat bahwa kurva perbandingan jumlah pipa termuat dengan waktu iterasi berbentuk kurva logaritmis. Hal ini memberikan pengertian bahwa untuk sedikit kenaikan pada jumlah pipa termuat yang besar akan membutuhkan waktu iterasi yang lebih lama. Hal ini terjadi karena pada jumlah pipa yang diuji semakin banyak dan jelas tempat-tempat yang perlu diuji juga semakin banyak. Maka kenaikan waktu iterasi menyerupai bentuk eksponensial.

Bila ditinjau dari sisi kekonsistensian jawaban, maka terlihat setidaknya dari 32 eksperimen yang dilakukan, jawaban yang diberikan adalah konsisten. Hal ini dapat dibuktikan dari eksperimen 5, 20 dan 28 (konfigurasi pengepakan ada pada Tabel 2). Eksperimen 5 merupakan parsial dari eksperimen 20 dan 28, tetapi menghasilkan jawaban yang sama walaupun dengan konfigurasi yang berbeda. Demikian pula dengan eksperimen 7, 24 dan 32 atau eksperimen 6, 19 dan 27 atau pada eksperimen 8, 23 dan 31. Kesemuanya walaupun memberikan konfigurasi yang berbeda namun dengan jumlah pipa termuat yang sama.

5. REVISI METODE SOLUSI

5.1 Kelemahan

Untuk eksperimen ke-26, 28, 30 dan 32 didapati satu fenomena yang unik. Hal itu tampak pada gambar berikut.

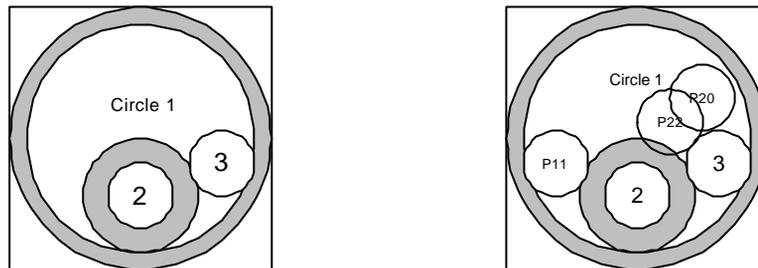


Gambar 3. Konfigurasi Pengepakan Silinder 90 mm di dalam Silinder 500 mm dan di atas Silinder 280 mm

Bila diperhatikan pada eksperimen ke 26, 28, 30 dan 32 yang melibatkan peletakan silinder 90 mm di dalam silinder 500 mm dan di atas silinder 280 mm selalu saja terbentuk konfigurasi yang demikian.

5.2 Penyebab Kelemahan

Penyebab kelemahan berasal dari angka posisi. Hal tersebut dapat diilustrasikan sebagai berikut. Misalkan telah ditempatkan 1 pipa besar (*circle 1*), 1 pipa sedang (*circle 2*) dan 1 pipa kecil (*circle 3*) seperti tampak dalam Gambar 4 bagian kiri.



Gambar 4. Ilustrasi Pendukung Penjelasan Kelemahan

Kemudian akan ditempatkan 1 pipa kecil (lingkaran keempat), seukuran dengan pipa 3, maka angka posisi yang terbentuk adalah seperti pada Tabel 3 berikut dengan penjelasan posisi seperti pada Gambar 4 bagian kanan.

Tabel 3. Angka Posisi dari Gambar 4 untuk Lingkaran Keempat

Lingka- Ran ke-	Kiri Atas	Kiri Bawah	Kanan Atas	Kanan Bawah	Menyen- tuh Dasar	Kiri Dalam	Kanan Dalam	Lingkaran Lain ke-	
								1	2
1	1	2	3	4	5				
2	6	7	8	9	10	11	12	13	
3	14	15	16	17	18	19	20	21	22

Pada Gambar 4 bagian kanan, tampak hanya ada 3 posisi saja yang mungkin untuk ditempati oleh lingkaran keempat. Bila diambil angka acak dari 1 hingga 22 maka peluang angka posisi 11 terpilih adalah $11/22 = 0.5$, peluang angka posisi 20 yang terpilih adalah $9/22 = 0.41$, peluang angka posisi 22 terpilih adalah $2/22 = 0.11$. Kecilnya peluang angka posisi 22 terpilih inilah yang menyebabkan sangat kurang kejadian terpilihnya peletakan pipa pada posisi semacam itu. Bila masih muncul kejadian diatas hal itu dikarenakan angka posisi 11 dan 20 memang tidak mungkin untuk ditempati pipa yang bersangkutan.

5.3 Analisa untuk Mengatasi Kelemahan

Untuk mengatasi kelemahan, tidak ada jalan lain kecuali memperbesar peluang terpilihnya posisi-posisi yang tidak seharusnya memiliki peluang yang kecil. Cara yang diusulkan disini adalah merubah sistem pembangkitan angka acak untuk tiap pipa. Cara yang sebelum ini digunakan adalah mengambil angka acak yang berada antara 1 hingga

$(i^2 + 11i - 16) / 2$ dan langsung melakukan pengujian peletakan pipa dimulai dari angka acak terpilih. Cara baru yang diusulkan adalah tetap dengan mengacak antara 1 hingga $(i^2 + 11i - 16) / 2$, kemudian dilakukan pengujian apakah pipa yang dijadikan acuan penempatan pipa baru berada didalam pipa lain. Apabila tidak berada di dalam pipa lain, maka proses akan dilanjutkan seperti semula. Karena kejadian kelemahan pertama hanya terjadi pada kondisi pipa di dalam pipa maka penanganan khusus diberlakukan apabila pipa acuan berada di dalam pipa lain. Penanganan khusus adalah dengan melakukan pengacakan yang baru dengan peluang yang sama untuk menentukan apakah pipa disusupkan (posisi 20 pada Gambar 4) atau diletakkan diantara pipa lainnya (posisi 22 pada Gambar 4). Dan disini peluang angka posisi yang tadinya kecil akan membesar karena memiliki peluang yang sama.

5. KESIMPULAN DAN SARAN

Permasalahan pengepakan pipa secara matematis diformulasikan sebagai permasalahan *non-linier mixed integer programming*. Karena formulasi yang demikian tersebut sulit untuk dipecahkan secara eksak maka digunakanlah metode heuristik. Metode heuristik yang digunakan adalah Algoritma Genetika yang menggunakan teori angka posisi sebagai bagian dekodisasi dari Algoritma Genetika umum dan penggunaan Daftar Tabu untuk mempercepat iterasi yang tak perlu.

Penggunaan angka posisi yang dimodifikasi pada tulisan ini dapat mengakomodasi semua tempat yang mungkin untuk ditempati oleh sebuah pipa. Angka posisi yang diusulkan disini mencakup keempat sisi luar dari lingkaran, berada di dalam dan di dasar lingkaran serta kemungkinan berada dalam lingkaran lain yang ditopang oleh sisi dalam lingkaran tersebut. Perubahan pada cara pembentukan permulaan angka posisi diperlukan untuk memperbesar peluang titik-titik posisi yang kecil. Pembesaran peluang memberikan hasil berupa titik-titik posisi yang tadinya tidak pernah terpilih dapat terpilih untuk ditempati oleh sebuah pipa. Perubahan hanya diberlakukan apabila pipa berada di dalam pipa acuan.

Algoritma Genetika secara umum menyebabkan terjadinya perbaikan nilai fungsi tujuan dibandingkan dengan individu-individu sebelumnya. Perbaikan nilai fungsi tujuan akan semakin baik bila angka acak yang dihasilkan untuk membentuk individu-individu pada Algoritma Genetika benar-benar acak, sehingga individu-individu awal yang dihasilkan berbeda satu dengan lainnya. Algoritma yang diusulkan pada tulisan ini mampu memberikan konfigurasi pengepakan pipa yang memuaskan bila ditinjau dari segi ruang yang tersisa dan ruang yang termanfaatkan, sehingga secara logika mampu memberikan jumlah pipa yang dapat dimasukkan ke kontainer lebih banyak, selain itu juga memberikan waktu komputasi yang cepat, bila ditinjau dari sisi waktu iterasi.

Dalam tulisan ini disarankan agar pengguna algoritma ini memperhatikan terlebih dahulu berapa waktu yang diperlukan untuk menyelesaikan satu individu sebagai patokan berapa kira-kira waktu yang diperlukan untuk menyelesaikan seluruh rangkaian algoritma. Tulisan ini menyajikan algoritma khusus untuk memecahkan permasalahan pengepakan pipa yang merupakan gabungan bagian pertama dan bagian kedua dari permasalahan pada pengepakan benda silindris. Untuk pengembangan lebih lanjut dari permasalahan pengepakan pipa ini, disarankan untuk memperhatikan kemungkinan perubahan dimensi dari pipa akibat panas atau akibat beban dari pipa lain, pemberian prioritas secara ekstrem pada jenis pipa tertentu ataupun kemungkinan penggunaan lebih dari satu kontainer dalam pengepakan pipa. Pengembangan teori ini dapat pula dilakukan

dengan memasukkan unsur keseimbangan berat. Semua hal ini dilakukan dengan tujuan agar teori ini lebih mudah diterapkan pada dunia industri.

DAFTAR PUSTAKA

- Bazaraa, M.S., 1979. *Nonlinear Programming : Theory and Algorithms*. John Wiley & Sons.
- Castillo, L, et al., 1998. Distribution network optimization : Finding the Most Economic Solution by Using Genetic Algorithms. *European Journal of Operational Research*, Vol. 108.
- Chen, C.S, et al, 1995. *An Analytical Model for Container Loading Problem*. European Journal of Operational Research, Vol. 80.
- Daya, Ben M, et al, 1998. A Tabu Search Approach for the Flow-Shop Scheduling Problem. *European Journal of Operational Research*, Vol 109.
- Dowland, K.A., 1991. Palletisation of Cylinders in Cases, *OR Spektrum*, Vol. 13.
- Dowland, K.A. and Dowland, W.B., 1990. *Optimal and Heuristic Solutions to Packing Problem - The State of the Art*. Working Paper EBMS.
- Dowland, K.A. and Dowland, W.B., 1992. Packing Problems. *European Journal of Operational Research*, Vol. 56.
- Fraser, H.J and George J.A., 1994. *Integrated Container Loading Software for Pulp and Paper Industry*. European Journal of Operational Research, Vol. 77.
- George, John A, et al., 1995. Packing Different Sized Circles Into a Rectangular Container. *European Journal of Operational Research* 84.
- Haessler, R.W., and Sweeney, P.E., 1991. Cutting Stock Problems and Solution Procedures. *European Journal of Operational Research*, Vol 54.
- Isermann, H., 1991. Heuristiken zur Lösung des Zweidimensionalen Packproblems für Rundgefäße. *OR Spektrum*, Vol. 13.
- Morton, Thomas E., 1993. *Heuristic Scheduling Systems*. John Wiley & Sons.
- Pinedo, Michael., 1995. *Scheduling : Theory, Algorithms and Systems*. Prentice Hall.
- Shang, Jen S, et al., 1993. *Multicriteria facility layout problem : An Integrated Approach*. European Journal of Operation Research Vol. 66.
- Valls, Vicente, et al, 1998. A Tabu search approach to machine scheduling. *European Journal of Operational Research*, Vol. 106.
- Wagner Bret J., 1999. *A Genetic Algorithm Solution for One-dimensional Bundled Stock Cutting*. European Journal of Operational Research, Vol. 117.