

12-3-2003

Application of Tabu Search Algorithm in Job Shop Scheduling

Betrianis Betrianis

Departemen Teknik Industri, Fakultas Teknik, Universitas Indonesia, Depok 16424, Indonesia,
betrianis@ie.ui.ac.id

Putu Teguh Aryawan

Departemen Teknik Industri, Fakultas Teknik, Universitas Indonesia, Depok 16424, Indonesia

Follow this and additional works at: <https://scholarhub.ui.ac.id/mjt>



Part of the [Chemical Engineering Commons](#), [Civil Engineering Commons](#), [Computer Engineering Commons](#), [Electrical and Electronics Commons](#), [Metallurgy Commons](#), [Ocean Engineering Commons](#), and the [Structural Engineering Commons](#)

Recommended Citation

Betrianis, Betrianis and Aryawan, Putu Teguh (2003) "Application of Tabu Search Algorithm in Job Shop Scheduling," *Makara Journal of Technology*. Vol. 7 : No. 3 , Article 5.

DOI: 10.7454/mst.v7i3.197

Available at: <https://scholarhub.ui.ac.id/mjt/vol7/iss3/5>

This Article is brought to you for free and open access by the Universitas Indonesia at UI Scholars Hub. It has been accepted for inclusion in Makara Journal of Technology by an authorized editor of UI Scholars Hub.

PENERAPAN ALGORITMA *TABU SEARCH* DALAM PENJADWALAN *JOB SHOP*

Betrianis dan Putu Teguh Aryawan

Departemen Teknik Industri, Fakultas Teknik, Universitas Indonesia, Depok 16424, Indonesia

E-mail: betrianis@ie.ui.ac.id

Abstrak

Tabu Search merupakan salah satu metode pemecahan permasalahan optimasi kombinatorial yang tergabung ke dalam *local search methods*. Metode ini bertujuan untuk mengefektifkan proses pencarian solusi terbaik dari suatu permasalahan optimasi kombinatorial yang berskala besar (bersifat *np-hard*), contohnya permasalahan penjadwalan *job shop*, dengan waktu komputasi yang relatif lebih kecil, namun tanpa ada jaminan akan tercapainya solusi yang optimal. Dalam penelitian ini, *Tabu search* diterapkan pada sebuah permasalahan penjadwalan *job shop* dengan tujuan untuk meminimalkan waktu proses total atau *makespan* (C_{max}). Penjadwalan menggunakan algoritma *Tabu Search* ini dilakukan terhadap tiga kasus, yaitu paket pesanan bulan September, Oktober dan Nopember, dimana untuk setiap paket pesanan dilakukan variasi terhadap *initial solution* dan panjang *tabu list*. Hasil penjadwalan ini kemudian dibandingkan dengan hasil penjadwalan lain yang menggunakan 4 macam metode *basic dispatching rules*, yaitu *Shortest Processing Time (SPT)*, *Earliest Due Date (EDD)*, *Most Work Remaining (MWKR)* dan *First Come First Served (FCFS)*. Hasil pengolahan data menunjukkan bahwa penjadwalan yang menggunakan algoritma *Tabu Search* sensitif terhadap perubahan yang diberikan pada variabel yang ada didalamnya dan *makespan* yang dihasilkan secara keseluruhan lebih kecil apabila dibandingkan dengan hasil penjadwalan menggunakan ke-4 metode lainnya.

Abstract

Application of Tabu Search Algorithm in Job Shop Scheduling. Tabu Search is one of local search methods which is used to solve the combinatorial optimization problem. This method aimed is to make the searching process of the best solution in a complex combinatorial optimization problem (*np hard*), ex : job shop scheduling problem, became more effective, in a less computational time but with no guarantee to optimum solution. In this paper, tabu search is used to solve the job shop scheduling problem consists of 3 (three) cases, which is ordering package of September, October and November with objective of minimizing makespan (C_{max}). For each ordering package, there is a combination for initial solution and tabu list length. These result then compared with 4 (four) other methods using basic dispatching rules such as Shortest Processing Time (SPT), Earliest Due Date (EDD), Most Work Remaining (MWKR) dan First Come First Served (FCFS). Scheduling used Tabu Search Algorithm is sensitive for variables changes and gives makespan shorter than scheduling used by other four methods.

Keywords : scheduling, tabu search, job shop, makespan

1. Pendahuluan

Dalam sebuah industri, baik itu industri manufaktur maupun jasa, adanya suatu proses penyusunan penjadwalan yang baik adalah sebuah hal yang penting. Hal ini dikarenakan dengan adanya penjadwalan yang baik akan dapat meningkatkan efektivitas serta efisiensi sistem produksi industri tersebut yang pada akhirnya akan mengurangi *production costs*. *Scheduling* dapat diartikan sebagai pengalokasian sejumlah *resources*

(sumber daya) untuk melakukan sejumlah *tasks* (tugas/operasi) dalam jangka waktu tertentu [1]. Secara umum *scheduling* merupakan suatu permasalahan dalam hal melakukan *sequencing* terhadap sejumlah operasi dan mengalokasikannya ke dalam slot waktu tertentu tanpa melanggar *technical constraints* (batasan teknis) dan *capacitive constraints* (keterbatasan kapasitas yang dimiliki). Baik secara teori maupun prakteknya di lapangan, untuk dapat melakukan suatu proses penjadwalan (*scheduling*) yang baik sangat sulit untuk

dibuat. Hal ini berdasar pada kenyataan bahwa begitu banyak parameter yang harus diperhatikan. Karena *scheduling*, khususnya *job shop scheduling*, merupakan suatu permasalahan *combinatorial optimization* yang kompleks maka permasalahan *scheduling* dapat dikategorikan sebagai permasalahan *np-hard*, yaitu suatu permasalahan yang pencarian solusinya (waktu komputasinya) akan naik secara eksponensial seiring dengan naiknya ukuran permasalahan secara linier [2]. Untuk itu diperlukan suatu metode yang lebih baik dalam memecahkan permasalahan ini.

2. Metode Penelitian

Salah satu metode penyelesaian permasalahan *np-hard* yang cukup efektif adalah metode algoritma heuristik (*Heuristic Algorithm*), yaitu suatu jenis algoritma yang termasuk ke dalam jenis algoritma sub-optimal. Meskipun algoritma heuristik termasuk ke dalam jenis algoritma suboptimal yang tidak dapat menjamin tercapainya suatu solusi yang optimal (*best solution*), namun algoritma heuristik akan dapat secara efektif mengatasi permasalahan *combinatorial optimization* yang cukup sulit dan berskala besar dengan cara mencari *good solution* yang dapat memuaskan semua kriteria dengan waktu komputasi yang relatif kecil, selain itu algoritma heuristik juga mudah diimplementasikan dan bersifat fleksibel. Salah satu metode algoritma heuristik yang cukup populer adalah *neighborhood search methods*, dimana didalamnya terdapat tiga macam metode, yaitu : *Genetic Algorithm*, *Simulated Annealing* dan *Tabu Search*. Ide tentang *Tabu Search* pertama kali diperkenalkan oleh Glover pada tahun 1970.

Konsep dasar dari *Tabu Search* adalah pengefektifan proses pencarian solusi dengan cara mencari *best solution* pada setiap tahap pelacakan [2-3]. Pada beberapa tahap pelacakan dapat dikategorikan sebagai langkah *tabu* (dilarang) karena akan menghasilkan *local optima* dan juga karena akan mengakibatkan langkah pengulangan kembali pencarian ke solusi yang pernah ditemukan sebelumnya (*entrainment*). Langkah-langkah ini kemudian dimasukkan ke dalam daftar yang disebut dengan *tabu list*. Proses pencariannya itu sendiri dilakukan dengan cara menentukan solusi awal dan kemudian dilakukan gerakan (*move*) ke solusi-solusi berikutnya (*neighborhood*) dan baru berhenti sampai kriteria penghentian (*stopping conditions*) tercapai. Metode ini sendiri telah digunakan pada berbagai bidang, diantaranya *resource planning*, telekomunikasi, penjadwalan, logistik, *space planning* dan sebagainya.

Penelitian berkaitan dengan *Tabu Search* telah banyak dilakukan. Chambers dan Barnes [4] menerapkan pendekatan *Tabu Search* yang dinamis, yang dapat disesuaikan dengan kondisi-kondisi pencarian yang berubah-ubah terutama pada sistem manufaktur yang

sangat fleksibel. Selain itu Ahmad, Dhodhi dan Ali [5] menerapkan *Tabu Search* untuk mendapatkan prioritas yang sesuai pada sintesa perilaku *Functional Pipeline* pada komputer.

Algoritma *Tabu Search*

Step 1

$k=1$

Select an initial schedule S_1 using some heuristic and set $S_{best}=S_1$

Step 2

Select $S_c \in N(S_k)$

If the move $S_k \rightarrow S_c$ is prohibited by a move on the tabu list

Then go to Step 2

If the move $S_k \rightarrow S_c$ is not prohibited by a move on the tabu-list

Then $S_{k-1}=S_c$

Enter reverse move at the top of the tabu-list

Push all other entries in the tabu-list one

position down

Delete the entry at the bottom of the tabu list

If $F(S_c) < F(S_{best})$ then $S_{best}=S_c$

Go to Step 3

Step 3

$k=k+1$

If stopping condition=true then STOP

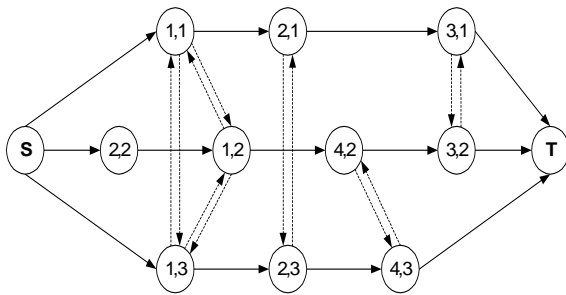
Else go to Step 2

(Sumber : Pinedo dan Chao, [1])

3. Pembahasan dan Hasil

Berikut ini karakteristik penjadwalan *Job-Shop* [1] :

- Ada sejumlah m mesin dan n job
- Setiap job terdiri dari satu rantai urutan operasi mengikuti rute yang telah ditentukan
- Setiap operasi dalam job diproses oleh salah satu mesin yang ada dengan waktu proses yang diasumsikan tetap
- Setiap operasi dapat melalui satu jenis mesin lebih dari satu kali (*recirculation*)
- Tidak boleh ada *preemption* (penundaan satu job oleh job yang lain)
- Fungsi tujuan permasalahan penjadwalan model *job shop* adalah untuk mencari satu jadwal yang meminimalkan *makespan*
- Permasalahan penjadwalan *job shop* merupakan permasalahan optimisasi kombinatorial yang kompleks (*np-hard*)
- Bentuk permasalahan penjadwalan model *job shop* dapat digambarkan ke dalam bentuk *disjunctive graph*, seperti berikut ini :



Gambar 1. Contoh permasalahan penjadwalan job shop dengan 4 mesin dan 3 job [1]

Keterangan :

- S,T = dummy
- i,j = proses job j pada mesin i
- conjunctive (solid) arcs = menunjukkan precedence relationship antar operasi
- disjunctive (broken) arcs = menghubungkan 2 operasi yang berasal dari 2 job yang berbeda yang harus diproses pada satu mesin yang sama

Secara garis besar prosedur umum yang diterapkan pada permasalahan penjadwalan dapat dibagi menjadi 2 kelompok [1], yaitu :

A. Constructive Procedure

Constructive procedure ialah suatu prosedur pemecahan permasalahan penjadwalan dimana solusi penjadwalan dibuat dalam satu kali proses pencarian sampai didapat satu solusi optimal yang lengkap. Metode yang termasuk kedalamnya antara lain :

- Basic Dispatching Rules
- Mathematical Programming
- Composite Dispatching Rules
- Branch and Bound
- Beam Search

B. Iterative Procedure

Iterative procedure berangkat dari satu solusi penjadwalan lengkap yang ditentukan secara acak atau dengan cara lain, yang kemudian solusi tersebut dimanipulasi secara bertahap untuk mendapatkan satu solusi yang optimal atau mendekati optimal.

- Classical Iterative Improvement
- Threshold Algorithms
- Tabu Search
- Simulated Annealing
- Genetic algorithms

Berdasarkan data waktu proses dan waktu set up tiap mesin, dapat dihitung waktu proses total untuk tiap operasi. Waktu total ini didapat dengan mengalikan waktu proses mesin dengan jumlah item yang dipesan dan kemudian ditambahkan dengan waktu set up dengan rumus :

$$P_{ijk} = (N_i \times p_k) + s_k \tag{1}$$

Dengan :

P_{ijk} = waktu proses total untuk operasi ke j pada job i yang menggunakan mesin k

N_i = jumlah item untuk job i yang dipesan

p_k = waktu proses mesin k

s_k = waktu setup mesin k

Perhitungan waktu proses untuk tiap operasi dan urutan rute prosesnya untuk tiap job untuk paket pesanan bulan September, Oktober dan November dapat dilihat pada gambar berikut ini :

Paket Pesanan Bulan September 1992

No. Item (Jml. Item)	Rute Proses (menit)																	
1 (5000)	1	10	7	3	10	6	8	10	284	296	157	122	296	574	630	296		
2 (8000)	2	10	4	10	9	10	578	474	474	474	1525	474						
3 (6000)	2	5	6	7	1	434	297	689	189	341								
4 (9000)	3	5	1	10	6	7	8	9	10	220	834	511	533	1033	283	1133	1715	533
5 (7000)	1	10	7	3	10	6	8	10	397	415	220	171	415	804	881	415		
6 (8000)	2	10	4	9	10	578	474	474	1525	474								
7 (9000)	10	3	7	8	10	533	220	283	1133	533								
8 (5000)	8	9	4	2	3	6	9	11	630	953	297	362	122	574	953	177		
9 (4000)	6	11	3	2	4	8	11	9	460	142	98	289	237	504	142	763		
10 (4000)	9	8	10	7	5	7	3	9	763	504	237	126	371	126	98	763		

Gambar 2. Perhitungan waktu proses tiap operasi untuk paket pesanan bulan September

Paket Pesanan Bulan Oktober 1992

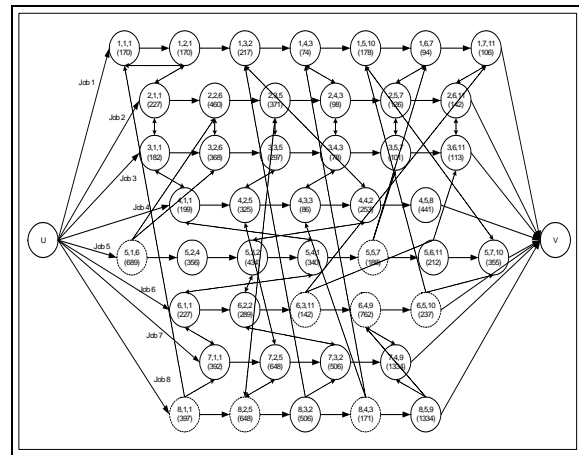
No. Item (Jml. Item)	Rute Proses (menit)													
1 (6000)	1	3	5	7	10	340	146	555	188	355				
2 (7000)	2	3	7	8	11	10	506	171	220	881	248	415		
3 (4000)	1	6	2	7	9	6	227	460	289	126	762	460		
4 (5000)	6	4	2	7	8	11	10	574	297	361	157	629	177	296
5 (5000)	6	4	2	1	7	11	10	574	297	361	284	157	177	296
6 (8000)	1	6	3	5	8	454	918	741	195	1007				

Gambar 3. Perhitungan waktu proses tiap operasi untuk paket pesanan bulan Oktober

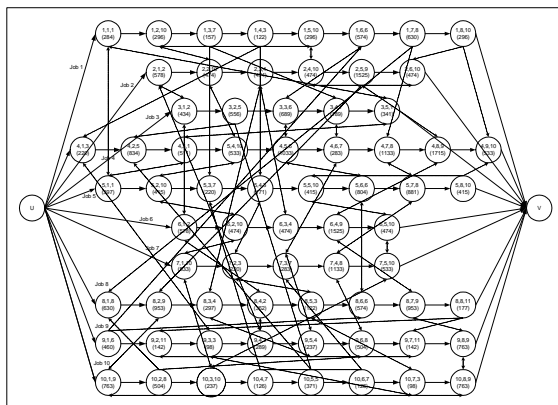
Paket Pesanan Bulan November 1992

No. Item (Jml. Item)	Rute Proses (menit)						
1 (3000)	1	1	2	3	10	7	11
	170	170	217	74	178	94	106
2 (4000)	1	6	5	3	7	11	
	227	460	371	98	126	142	
3 (3200)	1	6	5	3	7	11	
	182	368	297	78	101	113	
4 (3500)	1	5	3	2	8		
	199	325	86	253	441		
5 (6000)	6	4	2	1	7	11	10
	689	356	434	340	188	212	355
6 (4000)	1	2	11	9	10		
	227	289	142	762	237		
7 (7000)	1	5	2	9			
	397	648	506	1334			
8 (7000)	1	5	2	3	9		
	397	648	506	171	1334		

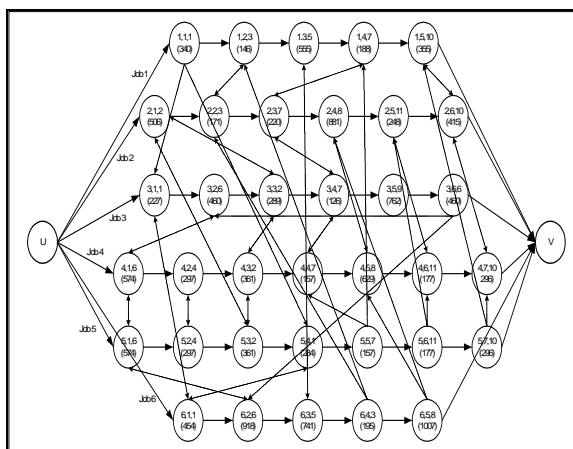
Gambar 4. Perhitungan waktu proses tiap operasi untuk paket pesanan bulan November



Gambar 7. Model Disjunctive Graph untuk Paket Pesanan Bulan November



Gambar 5. Model Disjunctive Graph untuk Paket Pesanan Bulan September



Gambar 6. Model Disjunctive Graph untuk Paket Pesanan Bulan Oktober

Selanjutnya permasalahan penjadwalan disajikan dalam bentuk model *Disjunctive Graph* untuk paket pesanan bulan September, Oktober dan November sebagaimana Gambar 7.

Penyelesaian penjadwalan dilakukan dengan menggunakan program *Pascal*. Untuk ketiga paket pesanan tersebut dilakukan penjadwalan dengan memakai 3 macam *basic dispatching rules* sebagai *initial solution*, yaitu *Earliest Due Date (EDD) job*, *Earliest Due Date (EDD) operasi* serta *Short Processing Time (SPT)*. Selain itu juga dilakukan kombinasi terhadap variabel *tabu tenur* sebanyak 5 macam yaitu 3, 5, 7, 9 dan 11 sehingga didapat 15 kombinasi penjadwalan untuk masing-masing paket pesanan. Hasil penjadwalan dapat dilihat pada Tabel 1.

Untuk penjadwalan paket pesanan bulan September menggunakan metode *Tabu Search*, terlihat bahwa hasil yang diberikan untuk ke-15 macam kombinasi penjadwalan cukup bervariasi. Hal ini dapat dilihat dari Tabel 1, dimana hasil terbaik diberikan pada penjadwalan yang menggunakan *EDD job* sebagai *initial solution* dengan *tabu tenur* 9, yaitu sebesar 9116 menit.

Untuk penjadwalan paket pesanan bulan Oktober, terlihat bahwa hasil yang diberikan untuk ke-15 macam kombinasi penjadwalan memberikan besar *makespan* yang sama, yaitu sebesar 4163 menit. Kecuali pada 3 macam penjadwalan yang memberikan hasil berbeda yaitu sebesar 4469 menit.

Untuk penjadwalan paket pesanan bulan November, terlihat bahwa hasil yang diberikan untuk ke-15 macam kombinasi penjadwalan kurang bervariasi. Seperti terlihat dari tabel di atas dimana penjadwalan yang

Tabel 1. Hasil penjadwalan paket pesanan bulan September (menit)

Tabu Tenur	Initial Solution		
	EDD job	EDD ops.	SPT
3	9779	9334	9974
5	9487	9157	9137
7	9487	9157	9137
9	9116	9157	9527
11	10197	9157	9527

Tabel 2. Hasil penjadwalan paket pesanan bulan Oktober (menit)

Tabu Tenur	Initial Solution		
	EDD job	EDD ops.	SPT
3	4469	4469	4163
5	4163	4163	4163
7	4163	4469	4163
9	4163	4163	4163
11	4163	4163	4163

Tabel 3. Hasil penjadwalan paket pesanan bulan November (menit)

Tabu Tenur	Initial Solution		
	EDD job	EDD ops.	SPT
3	4441	4441	5389
5	4441	4441	5389
7	4441	4441	5389
9	4441	4441	5389
11	4441	4441	5389

menggunakan *initial solution* EDD job dan EDD operasi memberikan besar *makespan* yang sama, yaitu sebesar 4441 menit, sedangkan penjadwalan yang menggunakan *initial solution* SPT memberikan *makespan* sebesar 5389 menit.

Secara keseluruhan analisa perbandingan hasil penjadwalan disajikan pada Tabel 4.

Tabel 4. Perbandingan Hasil Penjadwalan (menit)

Paket Pesanan Bulan September					
Metode	SPT	EDD	MWKR	FCFS	Tabu Search
Makespan	10624,2	11091,6	10475,4	10281	9116
Paket Pesanan Bulan Oktober					
Metode	SPT	EDD	MWKR	FCFS	Tabu Search
Makespan	5009,4	5554,2	5128,8	5009,4	4163
Paket Pesanan Bulan November					
Metode	SPT	EDD	MWKR	FCFS	Tabu Search
Makespan	6156,6	5332,8	5295	6426	4441

Dari tabel tersebut di atas dapat dilihat bahwa untuk keseluruhan paket pesanan, penjadwalan yang menggunakan metode *Tabu Search* memberikan hasil *makespan* yang lebih baik apabila dibandingkan dengan 4 (empat) metode lainnya, yaitu *Shortest Processing Time (SPT)*, *Earliest Due Date (EDD)*, *Most Work Remaining (MWKR)* dan *First Come First Served (FCFS)* [6].

4. Kesimpulan

Berdasarkan dari keseluruhan pembahasan mengenai proses optimalisasi penjadwalan *job shop* menggunakan metode *Tabu Search* di atas, maka dapat ditarik kesimpulan sebagai berikut :

Metode *Tabu Search* sensitif terhadap pemilihan *initial solution* serta panjang *tabu tenur*. Terlihat dari bervariasinya *current solution* dan solusi akhir (*best solution*) yang didapat untuk tiap-tiap *tabu tenur* pada setiap paket pesanan. Sebagai contoh paket pesanan bulan September, *best solution* yang dihasilkan cukup bervariasi tergantung dari *initial solution* serta *tabu tenur*-nya.

Untuk keseluruhan paket pesanan, penjadwalan yang menggunakan metode *Tabu Search* memberikan hasil *makespan* yang lebih baik apabila dibandingkan dengan 4 (empat) metode lainnya, yaitu *Shortest Processing Time (SPT)*, *Earliest Due Date (EDD)*, *Most Work Remaining (MWKR)* dan *First Come First Served (FCFS)*.

Daftar Acuan

- [1] M. Pinedo, X. Chao, Operations Scheduling with Applications in Manufacturing and Services, McGraw-Hill, Singapore, 1999.
- [2] M. Laguna, J. Barnes, F. Glover, Journal of International Manufacturing 2 (1991) 63.
- [3] B. Yen, G. Wan, Proceedings of ICOTA (1998) 1191.
- [4] John B. Chambers, J. Wesley Barnes, ORSA Journal 106 (1998) 254.
- [5] I. Ahmad, M.K. Dhodhi, F. M. Ali, Computer Journal 43 (2000) 152.
- [6] Riswan, Skripsi Sarjana, Jurusan Teknik Mesin, Fakultas Teknik, Universitas Indonesia, Indonesia, 1993.