

BAB 6

PENUTUP

6.1 Kesimpulan

Berdasarkan pembahasan yang sudah dipaparkan pada Bab 3–5, berikut kesimpulan-kesimpulan yang dapat diambil.

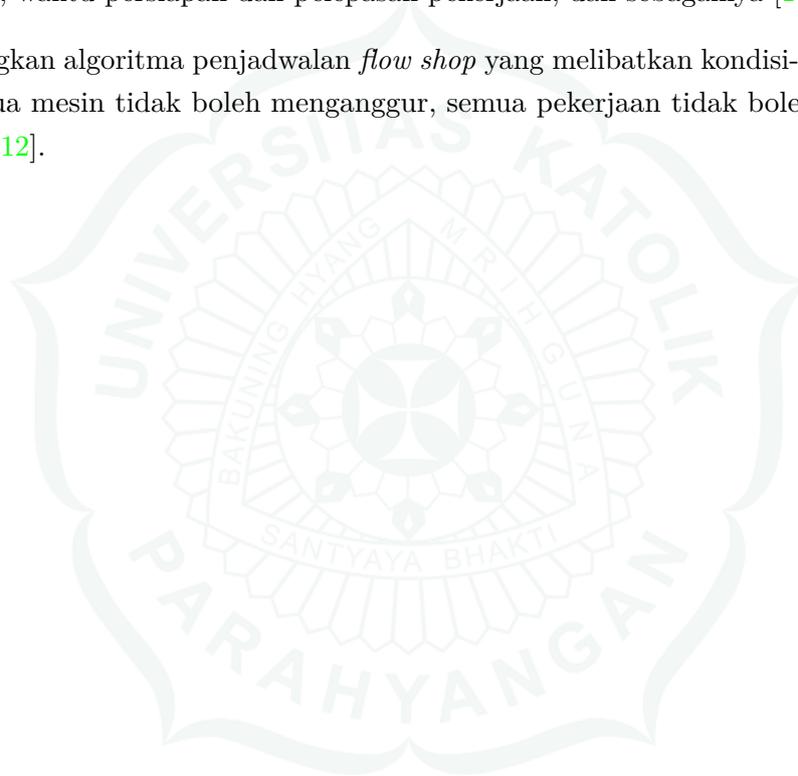
1. Semua masalah *flow shop* dua mesin dapat diselesaikan dengan algoritma sebagai berikut. Pertama, dibentuk dua himpunan U dan V , di mana U beranggotakan indeks-indeks pekerjaan yang durasi pengerjaannya di mesin pertama lebih cepat atau sama dengan durasi pengerjaannya di mesin kedua, sedangkan V beranggotakan indeks-indeks pekerjaan lainnya. Kedua, dibentuk dua barisan U^S dan V^S yang diperoleh dengan mengurutkan anggota-anggota himpunan U dan V . Anggota-anggota himpunan U diurutkan secara tak turun berdasarkan durasi pengerjaan di mesin pertama (dan untuk anggota-anggota dengan durasi pengerjaan di mesin pertama yang sama, diurutkan secara tak naik berdasarkan durasi pengerjaan di mesin kedua), sedangkan anggota-anggota himpunan V diurutkan secara tak naik berdasarkan durasi pengerjaan di mesin kedua (dan untuk anggota-anggota dengan durasi pengerjaan di mesin pertama yang sama, diurutkan secara tak turun berdasarkan durasi pengerjaan di mesin pertama). Konkatenasi dari kedua barisan U^S dan V^S adalah urutan yang optimal.
2. Penyelesaian masalah penjadwalan *flow shop* tiga mesin atau lebih bergantung pada banyaknya mesin yang digunakan.
 - (a) Masalah-masalah *flow shop* tiga mesin dapat diselesaikan dengan algoritma-algoritma yang telah dibahas jika memenuhi syarat-syarat cukup tertentu.
 - i. Syarat cukup yang pertama adalah nilai minimum dari semua durasi pengerjaan di mesin pertama lebih besar daripada nilai maksimum dari semua durasi pengerjaan di mesin kedua, atau nilai minimum dari semua durasi pengerjaan di mesin ketiga lebih besar daripada nilai maksimum dari semua durasi pengerjaan di mesin kedua. Jika syarat cukup tersebut terpenuhi, maka urutan yang optimal dapat diperoleh dengan langkah-langkah berikut. Pertama, kalikan durasi-durasi pengerjaan pada mesin pertama dan kedua, dan anggaplah durasi-durasi ini sebagai durasi-durasi pengerjaan pada mesin gabungan pertama. Kedua, kalikan durasi-durasi pengerjaan pada mesin kedua dan ketiga, dan anggaplah durasi-durasi ini sebagai durasi-durasi pengerjaan pada mesin gabungan kedua. Dengan menggunakan kedua mesin gabungan ini, dijalankan kembali algoritma penjadwalan *flow shop* dua mesin. Urutan yang dihasilkan adalah urutan yang optimal.

- ii. Syarat cukup yang kedua adalah nilai minimum dari semua durasi pengerjaan di mesin kedua lebih besar daripada nilai maksimum dari semua durasi pengerjaan di mesin pertama. Jika syarat cukup tersebut terpenuhi, maka urutan yang optimal dapat diperoleh dengan langkah-langkah berikut. Pertama, dibentuk urutan optimal dengan algoritma penjadwalan *flow shop* dua mesin pada mesin kedua dan ketiga. Kedua, tentukan urutan-urutan baru dengan memindahkan ke posisi pertama pekerjaan-pekerjaan yang memenuhi sifat bahwa durasi pengerjaan di mesin pertamanya lebih kecil daripada durasi pengerjaan pekerjaan pertama di mesin pertama pada urutan yang diperoleh pada langkah sebelumnya. Ketiga, hitung total durasi dari urutan-urutan yang terbentuk pada langkah pertama dan kedua. Urutan yang memberikan total durasi yang minimum adalah urutan yang optimal.
 - iii. Syarat cukup yang ketiga adalah nilai minimum dari semua durasi pengerjaan di mesin pertama lebih besar daripada nilai maksimum dari semua durasi pengerjaan di mesin ketiga. Jika syarat cukup tersebut terpenuhi, maka urutan yang optimal dapat diperoleh dengan langkah-langkah berikut. Pertama, dibentuk urutan optimal dengan algoritma penjadwalan *flow shop* dua mesin pada mesin pertama dan kedua. Kedua, tentukan urutan-urutan baru dengan memindahkan ke posisi terakhir pekerjaan-pekerjaan yang memenuhi sifat bahwa durasi pengerjaan di mesin ketiganya lebih kecil daripada durasi pengerjaan pekerjaan terakhir di mesin ketiga pada urutan yang diperoleh pada langkah sebelumnya. Ketiga, hitung total durasi dari urutan-urutan yang terbentuk pada langkah pertama dan kedua. Urutan yang memberikan total durasi yang minimum adalah urutan yang optimal.
 - (b) Masalah *flow shop* dengan $m > 3$ mesin yang dapat diselesaikan dengan algoritma yang telah dibahas hanyalah yang memenuhi syarat cukup bahwa nilai minimum dari semua durasi pengerjaan di mesin pertama lebih besar daripada nilai maksimum dari semua durasi pengerjaan di mesin kedua hingga $m - 1$, atau nilai minimum dari semua durasi pengerjaan di mesin terakhir lebih besar daripada nilai maksimum dari semua durasi pengerjaan di mesin kedua hingga $m - 1$. Dalam kasus ini, urutan yang optimal dapat ditentukan dengan langkah-langkah berikut. Pertama, kalikan durasi-durasi pengerjaan pada mesin pertama hingga $m - 1$, dan anggaplah durasi-durasi ini sebagai durasi-durasi pengerjaan pada mesin gabungan pertama. Kedua, kalikan durasi-durasi pengerjaan pada mesin kedua hingga terakhir, dan anggaplah durasi-durasi ini sebagai durasi-durasi pengerjaan pada mesin gabungan kedua. Dengan menggunakan kedua mesin gabungan ini, dieksekusi kembali algoritma penjadwalan *flow shop* dua mesin. Urutan yang dihasilkan adalah urutan yang optimal.
3. Waktu eksekusi program *trial and error* memang sedikit lebih singkat daripada waktu eksekusi program penjadwalan pada semua masalah penjadwalan *flow shop* untuk banyaknya pekerjaan yang kecil. Namun, apabila terdapat cukup banyak pekerjaan, maka waktu eksekusi program *trial and error* meningkat drastis sehingga program tersebut menjadi tidak efisien. Di lain pihak, dengan program penjadwalan, waktu eksekusinya tidak berbeda jauh dengan waktu eksekusinya untuk sedikit pekerjaan.

6.2 Saran

Karena berbagai keterbatasan penulis dalam mengeksplorasi berbagai kemungkinan pengembangan aljabar max-plus dalam masalah penjadwalan *flow shop*, skripsi ini masih memiliki peluang besar untuk dikembangkan. Beberapa saran yang diberikan untuk pengembangan adalah

1. membandingkan algoritma berbasis aljabar max-plus dengan berbagai algoritma penjadwalan *flow shop* yang lain seperti algoritma heuristik dan *branch and bound* [9, 10];
2. mengembangkan algoritma penjadwalan *flow shop* tiga mesin atau lebih dengan syarat-syarat cukup yang lebih lemah atau bahkan tanpa syarat cukup [3];
3. menambahkan faktor-faktor lain dalam penjadwalan *flow shop* untuk dianalisis, seperti waktu transportasi, waktu persiapan dan pelepasan pekerjaan, dan sebagainya [11];
4. mengembangkan algoritma penjadwalan *flow shop* yang melibatkan kondisi-kondisi tambahan, seperti semua mesin tidak boleh menganggur, semua pekerjaan tidak boleh menunggu, dan sebagainya [12].



DAFTAR REFERENSI

- [1] Šeda, M. (2007) Mathematical models of flow shop and job shop scheduling problems. *International Journal of Physical and Mathematical Sciences*, **1**, 307–312.
- [2] Johnson, S. M. (1954) Optimal two- and three-stage production schedules with setup times included. *Naval Research Logistics Quarterly*, **1**.
- [3] Bouquard, J. L., Lenté, C., dan Billaut, J. C. (2006) Application of an optimization problem in max-plus algebra to scheduling problems. *Discrete Applied Mathematics*, **154**.
- [4] Bellman, R., Esogbue, A., Nabeshima, I., dan Rodin, E. (2014) *Mathematical Aspects of Scheduling and Applications: Modern Applied Mathematics and Computer Science* International series in modern applied mathematics and computer science. Elsevier Science.
- [5] Kasana, H. S. dan Kumar, K. D. (2004) *Introductory Operations Research*. Springer Berlin Heidelberg.
- [6] Heidergott, B., Olsder, G. J., dan van der Woude, J. (2006) *Max Plus at Work: modeling and analysis of synchronized systems: a course on Max-Plus algebra and its applications*. Princeton University Press.
- [7] Johnsonbaugh, R. (2018) *Discrete Mathematics*, 8th edition. Pearson.
- [8] Fraleigh, J. dan Brand, N. (2020) *A First Course in Abstract Algebra*. Pearson Education.
- [9] Ho, J. C. dan Chang, Y.-L. (1991) A new heuristic for the n-job, m-machine flow-shop problem. *European Journal of Operational Research*, **52**, 194–202.
- [10] McMahon, G. B. dan Burton, P. G. (1967) Flow-shop scheduling with the branch-and-bound method. *Operations research*, **15**, 473–481.
- [11] Kubo, S. dan Nishinari, K. (2018) Applications of max-plus algebra to flow shop scheduling problems. *Discrete Applied Mathematics*, **247**, 278–293.
- [12] Žužek, T., Peperko, A., dan Kušar, J. (2019) A max-plus algebra approach for generating non-delay schedule. *Croatian Operational Research Review*, **10**, 35–44.