

ABSTRAK

Persaingan di dunia industri manufaktur semakin ketat dan suatu perusahaan untuk dapat bertahan harus dapat memenuhi kebutuhan konsumen, maka diperlukan suatu penjadwalan produksi yang tepat. Penjadwalan *flowshop* m mesin dengan memperhitungkan waktu *set up* yaitu penjadwalan *flowshop* pada beberapa mesin, dengan memperhatikan waktu persiapan yang terjadi saat pergantian operasi. Waktu *set up* yang dimaksud di sini adalah waktu *set up* yang tergantung pada urutan *job*. Tujuan meminimumkan jumlah *weighted flowtime* dan *weighted tardiness* dari *jobs* adalah tujuan penting pada banyak situasi kehidupan nyata, khususnya dengan memperhatikan tujuan untuk meminimumkan biaya selama *flowtime*, perjanjian biaya penalti keterlambatan pasokan dan penghargaan pelanggan. Oleh karena itu, tujuan ini menjadi tujuan yang lebih penting daripada meminimumkan *makespan*.

Penjadwalan *flowshop* m mesin dengan memperhatikan waktu *set up* tergantung urutan yang ada saat ini adalah penjadwalan pada *flowshop* yang statis. Dalam kenyataan sangat sering ditemui suatu kondisi yang semi dinamis. Sehingga dibutuhkan suatu algoritma penjadwalan produksi *flowshop* m mesin dengan memperhatikan waktu *set up* yang tergantung urutan pada *flowshop* semi dinamis.

Algoritma awal yang dianalisis dan menjadi acuan adalah algoritma yang dikembangkan oleh Rajedran dan Ziegler (2003) atau algoritma RZ, yang menghasilkan dua urutan, yaitu: urutan pertama yang mempertimbangkan *completion times* dari *job* yang ditambahkan dan *existing partial schedule*, selain itu juga mempertimbangkan biaya selama *flowtime* dan biaya *tardiness* dari *job* yang ditambahkan dan urutan kedua, dihasilkan dengan mempertimbangkan *due dates*, biaya selama *flowtime* dan biaya *tardiness* dari *job*. Yang terbaik dari kedua urutan tersebut akan dipilih untuk diterapkan *Improvement Scheme* sebanyak dua kali. Dilakukan pula uji coba algoritma CDS, yaitu algoritma yang meminimumkan *makespan* pada penjadwalan *flowshop*, untuk membuktikan bahwa meminimumkan *makespan* tidak sekaligus meminimumkan jumlah *weighted flowtime* dan *weighted tardiness* dari *jobs*.

Dari studi kasus yang dibuat, dapat disimpulkan bahwa algoritma RZ terbukti dapat menghasilkan urutan yang lebih meminimumkan jumlah *weighted flowtime* dan *weighted tardiness* dari *jobs* pada sebuah *flowshop* dengan waktu *set up* yang *sequence-dependent* dibanding algoritma CDS. Untuk studi kasus 7 job penghematan oleh algoritma RZ minimum 0.47% dan maksimum 15.60%. Pada studi kasus 10 job penghematan oleh algoritma RZ minimum 8% dan maksimum 14.92%. Untuk studi kasus 15 job, penghematan oleh algoritma RZ minimum 23.86% dan maksimum 34.31%. Algoritma usulan yang dibuat pada dasarnya menggunakan prinsip algoritma RZ. Hanya saja, ketika ada *job* tambahan yang datang, maka akan dilakukan penjadwalan ulang pada *job-job* yang belum diproses dan pada *job* tambahan tersebut. Selain itu, pada algoritma usulan ini, *improvement scheme* akan dilakukan terus sampai tidak menghasilkan perubahan hasil. Keterbatasan algoritma usulan ini adalah merupakan algoritma heuristik, karena itu tidak dapat menghasilkan penjadwalan yang optimum.

Analisis sensitivitas terhadap jumlah *job* awal, semakin banyak jumlah *job* awal, maka semakin kecil persentase selisih *makespan* dan persentase selisih jumlah *weighted flowtime* dan *weighted tardiness* dari *jobs*. Persentase selisih *makespan* terkecil (9.15%) dan persentase selisih jumlah *weighted flowtime* dan *weighted tardiness* dari *jobs* terkecil (33.28%) dihasilkan pada jumlah *job* awal terbanyak (15 *job*). Analisis sensitivitas terhadap jumlah *job* tambahan, semakin banyak jumlah *job* tambahan, maka semakin besar persentase selisih *makespan* dan persentase selisih jumlah *weighted flowtime* dan *weighted tardiness* dari *jobs*. Persentase selisih *makespan* terbesar (51.57%) dan persentase selisih jumlah *weighted flowtime* dan *weighted tardiness* dari *jobs* terbesar (167.23%) dihasilkan pada jumlah *job* tambahan terbanyak (5 *job* tambahan). Jumlah mesin mempengaruhi persentase selisih *makespan* dan persentase selisih jumlah *weighted flowtime* dan *weighted tardiness* dari *job*, perubahan jumlah mesin terhadap persentase selisih *makespan* dan persentase selisih jumlah *weighted flowtime* dan *weighted tardiness* dari *jobs* tidak membentuk suatu pola linier.

Kata Kunci: *flowshop*, semi dinamis, *weighted flowtime*, *weighted tardiness*.