ANALISIS KOMPLEKSITAS MASALAH PENJADWALAN SEMINAR ILMIAH

Lely Hiryanto[1], Tony[2], Dian Anggraini Cahyaningtyas[3]
[1,2,3] Fakultas Teknologi Informasi, Universitas Tarumanagara, Jln. Letjen S. Parman No. 1, Jakarta, 11440, Indonesia
E-mail: [1]lelyh@fti.untar.ac.id, [2]tony@fti.untar.ac.id, [3]dian.535190016@stu.untar.ac.id

Abstrak

Penjadwalan seminar ilmiah skala besar atau conference scheduling adalah salah satu masalah penjadwalan yang kompleks. Ada lima faktor utama yang dipertimbangkan ketika menyusun jadwal seminar ilmiah: (i) jumlah penyaji makalah, moderator dan pembicara tamu, (ii) kesediaan waktu moderator dan pembicara tamu, (iii) jumlah ruang seminar, (iv) jumlah sesi seminar dan (v) jumlah penyaji makalah yang dapat dijadwalkan dalam satu sesi seminar. Makalah ini menganalisis kompleksitas dari masalah seminar dengan mempertimbangkan kelima faktor tersebut. Analisis didasarkan pada penurunan dari tiga masalah yang telah terbukti memiliki kompleksitas Non-deterministic Polynomial Hard (NP Hard).

Kata kunci— *penjadwalan seminar, mathematical optimization, multiple knapsack problem, generalized multiple knapsack problem.*

Abstract

A scientific conference scheduling is one of the difficult and complex optimization problems. There are five main factors considered for a conference schedulling: (i) the number of paper presenters, moderators, and invited speakers, (ii) preference session of moderators and invited speakers, (iii) the availability of conference rooms, (iv) the number of sessions, and (v) the maximum presenters for each session. This article analyses the complexity of the conference scheduling that focuses on the aforementioned five main factors. It is done by comparing the scheduling problem with three well known problems that have been proven to be NP Hard.

Keywords—*conference Scheduling, mathematical optimization, multiple knapsack problem, quadratic multiple knapsack problem, generalized multiple knapsack problem.*

1. PENDAHULUAN

Seminar ilmiah atau sering disebut sebagai *conference* adalah sebuah pertemuan resmi sekelompok orang untuk mendiskusikan sebuah topik khusus atau topik umum yang sedang banyak diminati. Seminar skala besar atau sering diistilahkan conference melibatkan sejumlah besar pembicara dengan waktu pelaksanaan membutuhkan waktu satu hari penuh sampai satu minggu [1]. Selain itu, seminar dengan skala besar memiliki sejumlah sesi pembahasan topik khusus.

Permasalahan utama yang muncul dari proses penjadwalan seminar ilmiah berskala besar, yaitu (i) menjadwalkan para penyaji makalah ke setiap sesi yang sesuai dengan topik makalah mereka, dan (ii) menjadwalkan sejumlah sesi secara paralel yang menyesuaikan ketersediaan ruangan dan preferensi waktu dari moderator dan pembicara tamu (jika ada). Dua permasalahan tersebut menyebabkan penjadwalan seminar ilmiah menjadi kompleks dan sulit diselesaikan secara manual.

Multiple Knapsack Problem (MKP) adalah masalah pengalokasian sejumlah entitas ke sejumlah penampung yang memiliki kapasitas yang berbeda untuk memaksimalkan total keuntungan dari hasil pengalokasian [2]. Tiga kasus khusus dari MKP, yaitu *Generalized Assignment Problem* (GAP) dan *Quadratic Multiple Knapsack Problem* (QMKP) dan *Generalized Quadratic Multiple Knapsack Problem* (GQMKP), mempertimbangkan total keuntungan tersebut tidak hanya tergantung dari satu entitas, tetapi setiap pasang entitas yang dialokasikan ke setiap penampung [3]. Selain itu, GQMKP menyertakan preferensi penampung terhadap entitas.

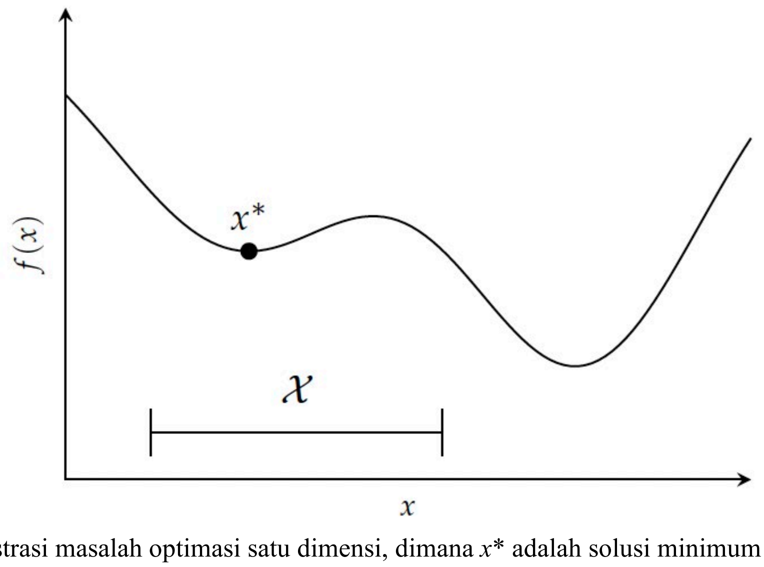
Tujuan dari artikel ini adalah untuk memodelkan secara matematis permasalahan penjadwalan seminar ilmiah dengan mengacu pada tiga kasus khusus dari MKP, yaitu GAP, QMKP, dan GQMKP. Model matematis dari permasalahan seminar tersebut selanjutnya di analisis kompleksitasnya berdasarkan pada ukuran dari permasalahan yaitu jumlah penyaji makalah, termasuk moderator dan pembicara tamu, serta jumlah sesi seminar yang tersedia.

2. METODE PENELITIAN

2.1 Model Matematika untuk Optimasi

Model matematika untuk permasalahan optimasi terdiri dari dua bagian yaitu [4]: (i) model matematika dan (ii) sekumpulan algoritma, atau *optimizer*, untuk menyelesaikan model matematika dan memberikan solusi optimal untik permasalah tersebut. Tingkat optimality dari solusi MO dipengaruhi oleh dua faktor yaitu akurasi dan probabilitas terkecil untuk menentukan satu solusi diantara banyak solusi optimal.

Struktur dari model matematika untuk optimasi terdiri dari: (i) target yang ingin dicapai (minimum atau maximum), (ii) variabel keputusan dan parameter, dan (iii) regulasi atau *constraints*. Gambar 1 menunjukan contoh masalah optimasi berdimensi satu, dimana titik solusi x^* bukanlah solusi minimum dari semua solusi global S, dimana $S \in X$.



Gambar 1. Ilustrasi masalah optimasi satu dimensi, dimana x^* adalah solusi minimum diantara semua kemungkinan solusi dalam himpunan X.

Titik solusi x merupakan sebuah vektor berdimensi n untuk n variabel keputusan yang berbeda. Titik solusi x berdimensi n dapat dipresentasikan sebagai himpunan variabel $[x_1, x_2, \ldots, x_n]$, dimana x_i menunjukkan variabel keputusan ke-i. Semua variabel di x dapat disesuaikan nilainya untuk meminimalkan fungsi target $f(x)$. Titik solusi optimal x^* memiliki nilai untuk semua variabel keputusannya yang meminimalkan fungsi $f(x)$ berdasarkan semua solusi *feasible* dalam himpunan X yang memenuhi sejumlah *constraints* dari masalah. Dalam hal ini, setiap titik solusi $x \in X$ memenuhi *constraints* yang telah ditetapkan. Untuk masalah optimasi yang targetnya adalah $f(x)$ maksimum dapat dimodelkan sebagai titik solusi optimal x^* yang memaksimalkan $-f(x)$.

2.2 Multiple Knapsack Problem

Multiple Knapsack Problem (MKP) memiliki banyak aplikasi untuk kasus-kasus nyata di lapangan seperti pengemasan barang ke dalam armada angkutan [5], pemotongan bahan (misalnya kerta, kain, dan besi) [6], dan penjadwalan ruang dan sesi operasi di rumah sakit [7]. Diketahui: (i) m penampung yang memiliki kapasitas c_i, dimana $i \in \{1,2,\ldots,m\}$, dan (i) n entitas dimana setiap entitas memiliki bobot w_j dan keuntungan p_j, dimana $j \in \{1,2,\ldots,n\}$ dan $m \leq n$. MKP bertujuan menentukan m sub-kelompok entitas yang memaksimalkan total keuntungan seperti disajikan di (1a).

Persamaan (1) berikut menyajikan model matematika dari MKP. Pada persamaan tersebut, total bobot setiap sub-kelompok ke-i tidak melebihi kapasitas c_i, seperti yang diperlihatkan oleh (1b) dan satu entitas dialokasikan ke tepat satu penampung yang dapat dilihat pada (1c). Berdasarkan (1d), variabel keputusan x_{ij} bernilai 1 jika entitas ke-j dimasukkan ke penampung ke-i, dan bernilai 0 jika sebaliknya. Parameter kapasitas c_i, bobot w_j, dan keuntungan p_i adalah bilangan bulat positif.

$$\underset{x_{ij}}{\text{maximize}} \quad \sum_{i=1}^{m} \sum_{j=1}^{n} p_j x_{ij} \tag{1a}$$

$$\text{subject to} \quad \sum_{j=1}^{n} w_j x_{ij} \leq c_i, \qquad i \in M = \{1,2,\ldots,m\}, \tag{1b}$$

$$\sum_{i=1}^{m} x_{ij} \leq 1, \qquad j \in N = \{1,2,\ldots,n\}, \tag{1c}$$

$$x_{ij} \in \{0,1\}, \quad i \in M, j \in N. \tag{1d}$$

2.2 Generalized Assignment Problem

Generalized Assignment Problem (GAP) mengamati masalah meminimalkan biaya penugasan n pekerjaan ke m agen, dimana satu pekerjaan ditugaskan ke tepat satu agen yang memiliki batasan jumlah pekerjaan yang bisa ditangani [8]. Perbedaan GAP dan MKP terletak pada keuntungan yang diperoleh oleh sebuah pekerjaan j ketika ditangani oleh agen i. Dalam hal ini, keuntungan dijadikan sebagai biaya penugasan, sehingga target dari GAP adalah meminimalkan total biaya penugasan. Persamaan (2) menyajikan model matematika dari GAP. Parameter p_{ij} di (2a) adalah biaya penugasan pekerjaan j ke agen i. *Constraint* (2b) memastikan bahwa setiap agen harus dialokasikan minimal satu pekerjaan.

$$\underset{x_{ij}}{\text{minimize}} \quad \sum_{i=1}^{m}\sum_{j=1}^{n} p_{ij}x_{ij} \tag{2a}$$

$$\text{subject to} \quad \text{(1b) and (1d)},$$

$$\sum_{i=1}^{m} x_{ij} = 1, \qquad j \in N. \tag{2b}$$

GAP banyak dipakai untuk aplikasi di dunia nyata, diantaranya yaitu [8]: (i) pengalokasian pembuatan aplikasi ke sejumlah programmer, (ii) penjadwalan iklan dengan berbagai durasi waktu di sejumlah slot waktu penayangan, dan (iii) penentuan lokasi pabrik.

2.3 Quadratic Multiple Knapsack Problem

Quadratic Multiple Knapsack Problem (QMKP) dapat diaplikasikan ke sejumlah masalah nyata, yaitu [9]: (i) pengalokasian anggota tim ke sejumlah proyek, (ii) penganggaran model, dan (ii) perancangan sistem pendistribusian barang. QMKP menangani masalah pengalokasian sepasang entitas, selain satu entitas, ke penampung untuk memaksimalkan total keuntungan [8]. Total keuntungan tersebut tidak hanya tergantung dari satu entitas, tetapi setiap pasang entitas yang dialokasikan ke setiap penampung. Model matematika untuk QMKP dapat dilihat di Persamaan (3) berikut ini.

$$\underset{x_{ij}}{\text{maximize}} \quad \sum_{i=1}^{m}\left(\sum_{j=1}^{n} p_j x_{ij} + \sum_{j=1}^{n}\sum_{k>j}^{n} q_{jk}x_{ij}x_{ik}\right) \tag{3}$$

$$\text{subject to} \quad \text{(1b)} - \text{(1d)}$$

Parameter q_{jk} merupakan keuntungan yang diperoleh jika entitas *j* dan entitas *k* berada di penampung yang sama. Target dari QMKP tidak hanya memaksimalkan keuntungan per entitas, tetapi juga keuntungan setiap pasang entitas di setiap penampung.

2.4 Generalized Quadratic Multiple Knapsack Problem

Generalized Quadratic Multiple Knapsack Problem (GQMKP) mengembangkan QMKP dengan menambahkan *constraints* terkait pengaturan dan preferensi penampung terhadap entitas [10]. Dalam hal ini, *n* entitas dikelompokkan ke dalam *r* kelas dengan setiap kelas memiliki waktu pengaturan yang berbeda. Selain itu, terdapat maksimal jumlah penampung dimana sejumlah entitas dari sebuah kelas dapat dialokasikan. GQMKP memiliki sejumlah aplikasi praktis diantaranya adalah penjadwalan produksi dan pabrikasi dimana pengaturan dan preferensi mesin harus dipertimbangkan [10].

GQMKP menyertakan enam parameter tambahan yaitu: (i) kumpulan entitas yang dapat dialokasikan ke setiap penampung, (ii) kumpulan kelas yang dapat diaktifkan oleh setiap penampung, (iii) kumpulan penampung dimana sebuah entitas dapat dialokasikan, (iv) kumpulan penampung dimana entitas dari sebuah kelas dapat dialokasikan, (v) waktu pengaturan, dan (vi) indikator yang bernilai 1 jika sebuah entitas termasuk di sebuah kelas.

Target dari GQMKP, seperti QMKP, adalah memaksimal total keuntungan. Tetapi, seperti GAP, keuntungan untuk setiap entitas ditentukan berdasarkan penampung dimana entitas tersebut dialokasikan. GQMKP memastikan total bobot dan waktu pengaturan dari entitas yang dialokasikan ke setiap penampung tidak melebihi kapasitasnya. Selanjutnya, setiap entitas dialokasikan minimal ke satu knapsack. Selain itu, jumlah penampung yang berisi entitas dari sebuah kelas tidak melebihi jumlah maksimalnya.

3. HASIL DAN PEMBAHASAN

3.1 Model Matematika Penjadwalan Seminar Ilmiah

Penjadwalan seminar ilmiah memiliki tiga macam entitas yaitu kumpulan penyaji makalah dinotasikan dengan N, kumpulan moderator dinotasikan dengan A, dan kumpulan pembicara tamu dinotasikan dengan B. Ketiga entitas tersebut tidak tergantung satu sama lain. Selain itu terdapat maksimal $m = |M|$ penampung yaitu sesi seminar. Setiap moderator dan pembicara tamu memiliki preferensi sesi seminar yang dapat mereka hadiri. Selain itu, setiap sesi memiliki maksimum total durasi waktu presentasi semua penyaji makalah dan alokasi ruang, dimana durasi atau slot waktu (bobot) untuk setiap penyaji makalah adalah sama. Dalam hal ini, diasumsikan setiap sesi $i \in M$ dapat menampung c_i makalah, dimana bobot w_j untuk setiap penyaji makalah $j \in N$ diset 1. Notasi R menandakan daftar ruang seminar yang tersedia.

Target dari masalah penjadwalan seminar adalah semua penyaji makalah dapat dijadwalkan disesi yang dapat dihadiri oleh satu moderator dan satu pembicara tamu (jika ada). Selain itu, setiap sesi diperuntukkan untuk penyaji makalah yang menyajikan topik pembahasan yang sama.

Diketahui r adalah kumpulan topik makalah yang berbeda; atau disebut juga sebagai kelas dari entitas. Berdasarkan karakteristik ini, masalah penjadwalan seminar memiliki kesamaan dengan *Quadratic Multiple Knapsack Problem* (QMKP), dimana keuntungan dari penjadwalan setiap penyaji makalah tergantung pada penampung (sesi) yang dialokasikan untuk penyaji makalah tersebut.

Penjadwalan seminar ilmiah, seperti QMKP, memaksimalkan total keuntungan tidak hanya tergantung dari satu entitas, tetapi setiap pasang entitas yang dialokasikan ke setiap penampung. Entitas dalam hal ini adalah penggabungan dari penyaji makalah, moderator, dan pembicara tamu. Selain itu, seperti GQMKP, keuntungan yang diperoleh oleh sebuah entitas j ketika dialokasikan ke penampung i akan berbeda jika entitas tersebut dialokasikan ke penampung lainnya. Hal ini didasarkan pada kesediaan waktu dari moderator dan/atau pembicara tamu.

Salah satu *constraint* yang perlu dipertimbangkan dan belum tercakup dalam QMKP, GAP dan GQMKP adalah pengalokasian entitas di sesi waktu seminar paralel, yaitu waktu dan hari yang sama dengan ruang seminar yang berbeda. Keterbatasan jumlah ruang menyebabkan sebuah seminar memiliki sesi paralel. Dua sesi dikatakan paralel jika pelaksanaannya di rentang waktu yang sama di ruang yang berbeda. Terkait hal ini, jika seorang penyaji makalah menjadi penyaji untuk dua makalah, maka penyaji makalah tersebut tidak bisa dialokasikan ke dua sesi paralel. Begitu pula dengan moderator dan pembicara tamu yang tidak bisa dialokasikan ke dua sesi paralel.

Dalam makalah ini, penulis mengembangkan QMKP menjadi *Extended* QMKP atau EQMKP. Varian baru ini, seperti QMKP, memaksimalkan total keuntungan tidak hanya tergantung dari satu entitas, tetapi setiap pasang entitas yang dialokasikan ke setiap penampung; lihat target di (4a). EQMKP menyertakan satu parameter tambahan yaitu h_{il} untuk menentukan jika sesi $i \in M$ paralel dengan sesi $l \in M$. Selain itu, EQMKP menambahkan dua variabel keputusan yaitu indikator y_{ia} yang menentukan apakah moderator $a \in A$ dialokasikan ke sesi $i \in M$ dan indikator z_{ib} untuk menandakan apakah pembicara tamu $b \in B$ dialokasikan ke sesi i.

$$\begin{aligned}
&\underset{x_{ij},y_{ia},z_{ib}}{\text{maximize}} && \sum_{i=1}^{m}\left(\sum_{j=1}^{n} p_{ij}x_{ij} + \sum_{j=1}^{n}\sum_{k>j}^{n} q_{jk}x_{ij}x_{ik}\right) && && \text{(4a)}\\
&\text{subject to} && \sum_{j=1}^{n} w_j x_{ij} \le c_i, && i \in M = \{1,2,\ldots,m\}, && \text{(4b)}\\
& && \sum_{i=1}^{m} x_{ij} \le 1, && j \in N = \{1,2,\ldots,n\}, && \text{(4c)}\\
& && \sum_{a\in A} y_{ia} = 1, && i \in M, && \text{(4d)}\\
& && \sum_{b\in B} z_{ib} <= 1, && i \in M, && \text{(4e)}\\
& && y_{ia} + y_{la} <= 1, && i,l \in M, l > i, h_{il} = 1, a \in A, && \text{(4f)}\\
& && z_{ib} + z_{lb} <= 1, && i,l \in M, l > i, h_{il} = 1, b \in B, && \text{(4g)}\\
& && 1 <= \sum_{i=1}^{m} z_{ib} <= max_b, && b \in B, && \text{(4h)}\\
& && x_{ij}, y_{ia}, z_{ib} \in \{0,1\}, && i \in M, j \in N. && \text{(4i)}
\end{aligned}$$

Constraint (4b) memastikan jumlah penyaji makalah dalam satu sesi tidak melebihi jumlah maksimum makalah per sesi. *Constraint* (4c) menjamin setiap penyaji makalah dialokasikan ke tepat satu sesi. Melalui *constraint* (4d), setiap sesi harus memiliki moderator, sedangkan setiap sesi bisa tidak memiliki pembicara tamu seperti yang diatur dalam *constraint* (4e). Selanjutnya, \ constraint (4f) dan constraint (4g) memastikan bahwa tidak ada seorang moderator dan seorang pembicara tamu dialokasikan di dua sesi paralel. *C*onstraint (4h), mengatur pengalokasian seorang seorang pembicara tamu tidak melebihi batas jumlah maksimal sesi dimana mereka bisa dialokasikan. Terakhir, *constraint* (4i) menentukan domain nilai dari variabel x_{ij}, y_{ia}, dan z_{ib} yaitu binary.

3.2 Analisis Kompleksitas Penjadwalan Seminar Ilmiah

MKP merupakan masalah optimasi dengan kompleksitas *Non-deterministic Polynomial* (NP) *Hard* yang membutuhkan waktu eksponensial untuk memecahkannya secara optimal [2]. Pembuktian oleh Martello and Toth [2] yang menggunakan 3-*partition* problem menunjukkan bahwa MKP tidak memiliki solusi dalam bentuk algoritma *pseudo-polynomial*. Dalam hal ini, MKP termasuk kategori *NP-hard in the strong sense*. Penambahan *constraint* sesi membuat penjadwalan seminar juga merupakan kasus khusus dari MKP. Oleh karena itu, penjadwalan seminar ilmiah dapat dibuktikan memiliki kompleksitas *NP-hard in the strong sense* dengan menggunakan pembuktian yang sama untuk MKP.

Berdasarkan Persamaan (4), kompleksitas dari permasalahan penjadwalan seminar ilmiah didominiasi dari jumlah penyaji makalah, jumlah moderator, jumlah pembicara tamu, dan jumlah sesi seminar yang tersedia. Persamaan (4) menggunakan tiga jenis variabel keputusan yaitu x_{ij}, y_{ia}, dan z_{ib}. Dapat dilihat pada Persamaan (4a), (4b), dan (4c), terdapat (mn) variabel keputusan untuk x_{ij} karena variabel tersebut digunakan untuk m sesi seminar dan n penyaji makalah. Untuk y_{ia}, Persamaan (4d) dan (4f) menggunakan secara total ($m|A|$) variabel keputusan. Selain itu, notasi z_{ib} pada Persamaan (4e), (4g), dan (4h) secara total berjumlah ($n|B|$). Total jumlah variabel keputusan adalah ($mn + m|A| + n|B|$).

Selanjutnya, terdapat delapan kelompok *constraints* pada Persamaan (4b) sampai dengan (4i). Kelompok *constraints* di Persamaan (4b), (4d), dan 4(e) mempertimbangkan semua sesi seminar sehingga terdapat 3m *constraints* secara total. Constraint di Persamaan (4c) untuk semua penyaji makalah, yaitu *n*. Untuk kelompok *constraints* di Persamaan (4f) dan 4(g), masing-masing dievaluasi maksimum sebanyak $\left(\frac{1}{|R|}m^2|A|\right)$ dan $\left(\frac{1}{|R|}m^2|B|\right)$. Kelompok *constraint* di Persamaan (4h) untuk (|B|) *constraints*. Terakhir, *constraint* di Persamaan (4i) memberikan nilai binary 0 atau 1 ke variabel keputusan x_{ij}, y_{ia}, dan z_{ib} sejumlah (*mn*). Jadi, total keseluruhan *constraints* adalah$\left(3m + n + \left(\frac{1}{|R|}m^2|A|\right) + \left(\frac{1}{|R|}m^2|B|\right) + mn\right)$.

Untuk catatan, $|A| < m$, $|B| < m$, dan $|B|<|A|$. Oleh karena itu, jumlah variabel keputusan dan jumlah total constraint, waktu eksekusi untuk mencari kombinasi penyaji makalah, moderator dan pembicara tamu yang optimal meningkat sejalan dengan bertambahnya jumlah penyaji makalah dan banyaknya sesi seminar.

4. KESIMPULAN

Masalah penjadwalan seminar ilmiah dan *Multiple Knapsack Problem* memiliki memiliki kemiripan karakteristik, yaitu: (i) entitas adalah penyaji makalah, moderator, dan pembicara tamu, (ii) penampung adalah sesi waktu seminar di hari dan ruang yang berbeda, dan (iii) bobot adalah maksimal durasi waktu presentasi penyaji makalah. Tujuan dari penjadwalan seminar tersebut untuk memaksimalkan total keuntungan dari setiap pasang entitas yang dialokasikan ke setiap penampung dan keuntungan yang diperoleh oleh sebuah entitas *j* ketika dialokasikan ke penampung *i*. Berdasarkan karakteristik tersebut, masalah penjadwalan seminar ilmiah memiliki kesamaan dengan QMKP, yang seperti MKP, termasuk dalam masalah *NP-Hard in the strong sense*. Akan tetapi, QMKP dikembang lebih lanjut menjadi EQMKP yang menyertakan *constraint* tambahan yaitu penanganan masalah sesi paralel, yaitu dua atau lebih sesi dengan slot waktu sama tetapi berbeda ruang, untuk pengalokasian penyaji makalah, moderator, dan pembicara tamu. Hasil analisis kompleksitas waktu juga meningkat seiring dengan bertambahnya jumlah penyaji makalah dan sesi seminar.

Penelitian selanjutnya akan memfokuskan pada implementasi model matematika dari masalah penjadwalan seminar ilmiah menggunakan optimizer tools, misalnya Gurobi Optimizer [11], untuk mendapatkan solusi optimal dari sejumlah sampel kegiatan seminar ilmiah. Selain itu, dikarenakan penjadwalan seminar ilmiah adalah termasuk masalah *NP-Hard*, akan diusulkan sejumlah algoritma *greedy-based heuristic* dan *metaheuristic* untuk mendekati solusi optimal tapi dengan waktu eksekusi yang lebih cepat.

DAFTAR PUSTAKA

[1] Bulhoes, T., Correia, R. and Subramanian, A., 2022, Conference scheduling: A clustering-based approach, European Journal of Operational Research, vol. 297, no. 1, Hal. 15–26, 2022.

[2] Martello, S. and Toth, P., 1990, Knapsack problems: algorithms and computer implementations, John Wiley & Sons, Inc.

[3] Cacchiani, V., Iori, M., Locatelli, A. and Martello, S., 2022, Knapsack problems-an overview of recent advances. part II: Multiple, multidimensional, and quadratic knapsack problems, Computers & Operations Research, Hal. 105693.

[4] Kochenderfer, M. J. and Wheeler, T. A., 2019, Algorithms for optimization. MIT Press.

[5] Dell'Amico, M., Delorme, M., Lori, M. and Martello, S., 2019, Mathematical models and decomposition methods for the multiple knapsack problem, European Journal of Operational Research, vol. 274, no. 3, pp. 886–899.

[6] Pisinger, D., 1999, An exact algorithm for large multiple knapsack problems, European Journal of Operational Research, vol. 114, no. 3, Hal. 528–541.

[7] Agnetis, A., Coppi, A., Corsini, M., Dellino, G., Meloni, C. and Pranzo, M., 2014, A decomposition approach for the combined master surgical schedule and surgical case assignment problems, Health care management science, vol. 17, no. 1, Hal. 49–59.

[8] Cattrysse, D. G. and Van Wassenhove, L. N., 1992, A survey of algorithms for the generalized assignment problem, European journal of operational research, vol. 60, no. 3, Hal. 260–272.

[9] Fleszar, K., 2022, A branch-and-bound algorithm for the quadratic multiple knapsack problem, European Journal of Operational Research, vol. 298, no. 1, Hal. 89–98.

[10] Zhou, Q., Hao, J.-K. and Wu, Q., 2022, A hybrid evolutionary search for the generalized quadratic multiple knapsack problem, European Journal of Operational Research, vol. 296, no. 3, Hal. 788–803.

[11] Gurobi Optimization, 2022, "Gurobi optimizer," https://www.gurobi.com/products/gurobi-optimizer/, accessed: Aug 2022.