Journal of Fundamental and Applied Sciences

ISSN 1112-9867

Available online at http://www.jfas.info

Research Article

Special Issue

INCORPORATING CAPACITATIVE CONSTRAINT TO THE PREFERENCE-BASED CONFERENCE SCHEDULING VIA DOMAIN TRANSFORMATION APPROACH

S. N. Jaafar[1,*], S. K. N. A. Rahim[1], N. Omar[1], S. Masrom[2] and N. Khairudin[2]

[1]Faculty of Computer and Mathematical Sciences, UniversitiTeknologi MARA, 40450 Shah Alam, Selangor, Malaysia

[2]Faculty of Computer and Mathematical Sciences, UniversitiTeknologi MARA, 35400 Tapah, Perak, Malaysia

Published online: 17 October 2017

ABSTRACT

Organizing academic conference often creates conflicts among presenters in the form of unsatisfactory assignments of papers to time slots or sessions during the conference. This paper proposes an enhanced technique of scheduling the conference papers to time slots or sessions by applying Domain Transformation Approach (DTA). The objective of this work is to generate a conference schedule that satisfies hard constraint which ensures all papers associated with presenters and participants are scheduled without conflicts, besides balancing the numbers of papers in the overall slots. According to the results, the schedules generated are of high quality which managed to satisfy the constraints imposed. Based on our findings, DTA produced competitive schedules efficiently as expected which satisfied all of the participants' preferences and balancing the allocation of papers to time slots.

Keywords: conference scheduling; domain transformation approach; capacity optimization.

Author Correspondence, e-mail: nazehajaafar@gmail.com

doi: http://dx.doi.org/10.4314/jfas.v9i5s.29

1. INTRODUCTION

From the simplest scheduling such as organizing tasks to be done every day to the more complex scheduling for example scheduling the university examinations papers or transportation scheduling, it was observed that scheduling is inevitable in one's daily life. As a result, there has been constant interest and development done to solve all kinds of scheduling problems. In this study, we are focusing our effort in enhancing the conference scheduling.

A conference is an academic event where a group of people meet, discuss, debate and confer about a certain topic or field [1]. In a conference, normally presenters will present their paper(s) or research and receive comments and suggestions from other presenters or conference participants who have interests in the research area. All researchers attending will benefit, as they exchange ideas and knowledge with each other. Thus, participants usually prefer to attend presentations of papers based on their area of interests or favourite fields and normally many of them expect the conference schedule can cater all of their preferences [2].

In a conference, presentation papers will need to be scheduled into time slots or sessions by the organizer [3] for the conference to run smoothly and to satisfy at least its hard constrains which is scheduling all the papers within the conference period. The current trend nowadays is to also satisfy all participants' preferences in terms of the paper presentation that they are able to attend [2]. The preferences include the sessions of interest, allocation of time in the timetable and room capacity planning. The current trend of fulfilling participants preference is due to the situation where participants feel unsatisfied [4] with the conference schedule, where the organizing committee randomly construct a schedule without taking into account the attendee's area of interest [1]. In addition, several sessions of presentations are scheduled on the same exact time with each other (parallel sessions) with the aim to have less duration of the conference and enabling more papers to be presented within the allocated time. Parallel sessions cause some of the presentations preferred by the participants overlap with one another, in fact they were unable to attend their sessions of interests.In fact, they were unable to attend all their sessions of interests, resulting in the participants feeling regret and feeling that the conference is not worth attending.

2. BACKGROUND OF THE STUDY

2.1. Some Types of Scheduling and Related Work

There are a few types of scheduling being investigated extensively by researchers. According to the scheduling literature, the most widely researched scheduling area is the educational scheduling which includes university scheduling, course scheduling, class-teacher scheduling, etc.

Usually, out of the many types of educational scheduling mentioned, scheduling the university timetable has been the famous topic of interests among researchers [2, 4] in which both course scheduling [5-8] and examination scheduling [4, 9-13] are being given a lot of attentions. Several solutions have been proposed to solve the university scheduling problem. For example, a mathematical model that used binary programming [5] with the objective to maximize education quality. The model used will assign each course with the most suitable or eligible lecturer. Other researches are using different methods such as graph coloring heuristic, local search technique, hybridizations, etc. where mostly produced promising findings and results based on its various evaluated aspects.

Apart from educational scheduling, other areas are also being studied extensively which include the transportation scheduling, staff (rostering) scheduling, nurse scheduling, sports scheduling and many more due to the fact that scheduling is a tedious and complex process, and is very time consuming if it is done manually. Therefore, there is a great need in automating the process, finding ways in generating schedules efficiently and creating the best schedule fulfilling the most of the constraints.

2.2. Hard Constraint and Soft Constraint

There are two types of constraints that are well known in the timetabling or scheduling area. First type of constraint is the hard constraint in which when generating a schedule, hard constraint must be satisfied at all times in order to produce a feasible timetable; violating any one of the constraints makes the schedule non-practical. Hence, it is crucial to achieve a feasible timetable that has no conflict or clashes. The second constraint is called the soft constraint where this constraint is optional to be satisfied, but it will make the schedule more appropriate and have better quality without affecting the hard constraint.

In this study, we are aiming at generating a feasible schedule by satisfying the hard constraint and to maximize the satisfaction of participants as well as ensuring that the number of papers assigned to a slot does not violate the time constrains.

2.3. Conference Scheduling

Conference scheduling usually involves many factors that play an important role in ensuring a smooth conference. For example, aspects that we need to take into accounts are the number of days needed to run the conference, the number of required sessions or time slots and the number of rooms for presentations [2].

In designing the conference scheduling, it is important to cater the participants' preferences to maximize their satisfactions in attending the sessions of interests. The numbers of papers per slots or sessions is also important, where normally the same range of numbers of papers per slot is preferable. This is with the objective to balance the number of presentations in each session, thus durations for each session can be standardized.

Other criteria that are being considered to produce better conference schedule are to group the papers according to the same research key-words, minimize session hopping and allocate enough space for participants to attend a particular presentation.

2.4. State of the Art

From the literature, it shows that the research trend for conference scheduling is moving into generating a schedule that is based on participants' preferences [1-3, 14-16]. Many approaches and methods have been used to improve the quality of the conference schedule, such as using goal programming [1], bi-criterion approach [2] that were inspired from Sampson's PBCS [3], integer programming [16], granular processing and Domain Transformation Approach (DTA) [17-18] and many more. It was reported that these methods produced very encouraging results.

2.5. Capacitated Constraint

In the literature, there is also research on optimization capacity planning. When considering the capacity problem, there are a few methods have been used in the literature. For example, goal programming [1], bin packing [17-18], heuristic, etc.

Based on [19], Bin Packing has been widely used in solving problems with regards to optimization of capacity constraint due to the good results produced. Thus, in this study, we are implementing Bin Packing approach for our capacity constraint. There are three types of methods under Bin Packing approach, which is First Fit, Next Fit and Worst Fit. For our study, we are using First Fit method because according to [20], First Fit is fast in allocating the papers to slot and has less possibility in adding a new slot.

3. MOTIVATIONS AND OBJECTIVES

This study was motivated by the problem to schedule the SimposiumKebangsaan Sains Kuantitatif (SKSK06) conference papers done by [1], where presenters and participants were given questionnaire to choose and rank 10 papers they would like to attend its presentations. The problem is known as preference-based conference scheduling problem, due to the considerations of preferences made by the participants.

Based on these preferences, the organizer will schedule the conference papers that can satisfy all the choices made by the presenters and participants without any conflict. In [1] managed to generate feasible schedule using goal programming in which it successfully scheduled the papers into sessions by satisfying preferences made, where up to 3 papers chosen by participants were satisfied. In our study, we are looking at improving the quality of the schedule by satisfying all preferences.In other words, scheduling all papers by considering all 10 preferences made by the respondents without any conflicts.

Besides that, this study is also motivated by the research done in [21] where encouraging results were obtained through Domain Transformation Approach(DTA). Despite the good schedule generated which managed to satisfy all preferences, the number of papers allocated to each slot is not balanced. It is important to limit and even out the number of papers per slot so that only certain numbers of papers are assigned to a slot. This is because each slot has a limited time of period allotted [22] and it is essential for the organizer to ensure that all papers get the chance to be presented, and the durations allocated are adhered. Inspired to produce better quality schedule, this study is also looking at the possibility of balancing out the number of papers in each slot as highlighted by [18].

In brief, the objective of this study is to generate a feasible schedule on preference-based conference scheduling problem with capacitated. The timetable will be enhanced so that all papers can be scheduled to session or slot without any conflict, ensuring its feasibility where all participants and presenters preferences are satisfied. We are also aiming at balancing the numbers of papers in each slot to improve the quality of the schedule.

To achieve all the objectives, this study is applying the DTA designed for Exam Scheduling Problem (ESP) proposed in [4], that has proven to be very reliable in producing promising results in generating the university’s examinations schedule [4, 13, 21, 23-26]. We adopt the same approach to test the flexibility of the approach either it is applicable to other area for scheduling or not. Besides, DTA has optimization process involve in its framework where the procedure in this stage can be substituted to solve the capacitated problem.

However, in practice, several other issues need to be considered in order to consummate the satisfaction of conference's participants. Not only maximizing the satisfaction of participants and generating a feasible schedule, we are also focusing on improving the capacity problem for papers allocated per time slot. Further explanations with regards to the methods used will be given in the following section.

4. METHODOLOGY

4.1.Domain Transformation Approach (DTA)

Domain Transformation Approach (DTA) can be defined as an approach that solved problem in a simpler structure as it breaks the problem into several stages [4]. This makes the bigger problem becomes less complex and solvable. From the literature, it was stated that DTA managed to generate a deterministic pattern of schedule in the exam scheduling problem [4, [12-13] and also encouraging results in the nurse scheduling problem [24-26]. Recently, DTA has proven its flexibility and generality by successfully generating good conference schedule [21]. DTA designed for examination scheduling problem was used to solve the preference-based conference scheduling problem, and as expected good quality schedule was generated. Inspired by the robustness of DTA, in this study we will be employing the following DTA framework to solve the preference-based conference scheduling problem:

- Datasets retrieval and standardization.
- Pre-processing the data
- Schedule through the conflict chain.
- Optimization-to even out the capacity of paper per slot.

1) Dataset retrieval and standardization: In this step, dataset will be retrieved and organized into a standard form. This is important because there is no standard way to represent the conference scheduling problem. This stage will ensure the dataset are ready for pre-processing to obtain more meaningful data.

2) Pre-processing the data: In this step, we pre-process the standardized form of dataset to generate the conflict matrix. Conflict matrix is a square matrix of dimension equal to the number of papers in the problem. Value 1 in the matrix at position (i,j) represents that the paper i is clashing with paper j. The conflict matrix is generated by incrementing the value at position (i,j) by 1 for each participant selecting to view paper i and j when the dataset is

traversed. The conflict matrix is a static data representation of the problem space. Information contain in the matrix is fixed, which represent the interrelation between paper selected for viewing by a participant.

3) Schedule through the conflict chain:In this step, we will undergo the conflict chain generation process using the data derived from the conflict matrix. In generating the conflict chain, papers which are conflicting are grouped together in a chain, and are assigned to its respective time slot. The length of the chain can be measured as the number of minimum time slots needed to schedule all the papers. It is worth highlighting that interestingly generation of the conflict chain is a very worthwhile process. This is because at this point we can consider the scheduling phase is done since papers are already assigned to a minimum number of time slots. Of course, at this particular stage, capacitative constraint is not yet being considered as it will be solved during the optimization stage. For further understanding, Fig. 1 shows the algorithm of the conflict chain generation.

1. Initiate the algorithm by allocating all papers to time slot one.
2. Select the first paper as “current” and initiate the counter for the current conflict chain.
3. Label the current paper as “allocated to the current chain” and note all of the papers that are in potential conflict with the current paper.
4. If the list of potentially conflicting papers is non-empty, re-allocate those papers to the next available time slot. Otherwise, label the current chain as complete and proceed to Step 6.
5. If the list of potentially conflicting papers is non-empty, select the first paper from the list and repeat from Step 3 with the currently selected paper.
6. Check if all papers allocated at Step 1 are belonging to one of the conflict chains; if YES, then the algorithm terminates; if NO, then the conflict chain counter is incremented and the unallocated paper is taken as “current” for processing, starting from Step 3.

Fig.1. Algorithm of conflict chain

4) Optimization-even out the capacity of paper per slot: In this last step, we will optimize the schedule generated earlier to ensure the allocation of papers to slot is within the allowed number. We used a different method from the original DTA because of the different subject structure. In [4] use Hill Climbing (HC) strategy and Late Acceptance Hill Climbing (LAHC) strategy to minimize the schedule cost once a feasible schedule has been generated. We use

one of the Bin Packing approach which is First Fit method incorporated with some rules to accomplish a capacitated schedule. First Fit scenario puts each respective paper into the first time slot that fits without any conflict. Fig. 2 shows the algorithm of the optimization process that incorporates the First Fit method.

1. Set the desired maximum number of paper per time slot.
2. Check either any of the time slots exceed the maximum number of paper per time slot; if YES, proceed to Step 3; if NO, algorithm terminate.
3. Find the time slot that exceed the maximum number of paper per time slot. Hold the last paper's ID of that particular time slot, paper X.
4. Find the time slot with the lowest number of paper, time slot Y.
5. Compare each paper's ID in time slot Y with the paper X. Check either they are conflict with each other or not; if NO, then place paper X into time slot Y (Bin Packing, First Fit applied) and start again with the step 2; if YES, find the next time slot with the lowest number of paper and repeat this step again. If paper X cannot fit into any of the time slots due to conflict with other paper, take the previous paper's ID in the same time slot of paper X and repeat step 5.

Fig.2. Algorithm for capacity optimization

In [17-18, 27-28], Bin Packing has been used by many for grouping to solving the capacitated problem. Instead of using Next Fit or Worst Fit i.e. other methods that lie under Bin Packing, First Fit to the best of our knowledge is more suitable to be used in our case. Based on [20], definition of first fit method is it will put each item (paper) as you come to it into the available slot (slots created from the previous process) into which it fits (has no conflict).

According to [29, 20], the definition in allocating the conference paper is next fit will start the allocation process on where the previous searching is stopped and never go back and if there is no slots suitable slots available, it will open a new slot. Comparing First Fit with other two methods, Next Fit for example, it will not only ignore the slots the visited slots from before to schedule the respective paper, it will also add more slots if the paper is not fitted anywhere in the available slots, this will result in more cost in conducting the conference scheduling as the time of the conference held will be longer due to the newly created slots. While Worst Fit will schedule it to the most conflicting slot found and this is not suitable for this scheduling.

Hence, the First Fit method is applied.

5. EXPERIMENT

In this section, we will show the actual data that we used to conduct this study and some results that we obtained after experimenting with the DTA in generating a feasible schedule.

5.1. Dataset

The data that we used is the real data that were obtained from SKSK06 conference. Participants were provided with a questionnaire when they register as a presenter or participant before the conference started. The questionnaire asked the participants to choose 10 papers that they were interested in attending or viewing the presentations (excluding the paper that they will be presenting themselves, if any of the participants happen to be a presenter).

Table 1. List of papers preferred by participants and presenters

ID	Paper's ID	ID	Paper's ID	ID	Paper's ID	ID	Paper's ID
1	1, 3, 42, 33, 21, 19, 49, 47, 8, 23, 9	21	21	41	41	61	42, 23, 41, 5, 15, 9, 57, 43, 48, 31
2	2, 12, 5, 13, 57, 47, 49, 58, 60, 48, 55	22	22	42	42	62	51, 24, 2, 11, 55, 1, 17, 12, 57, 60
3	3, 9, 11, 22, 44, 26, 24, 15, 51, 25, 10	23	23	43	43	63	47, 24, 29, 48, 57, 32, 23, 15, 55, 3
4	4, 7, 12, 20, 5, 49, 10, 43, 31, 26, 6	24	24	44	44	64	46, 50, 21, 41, 56, 54, 51, 18, 14, 20
5	5, 11, 10, 7, 25, 36, 15, 0, 30, 32, 12	25	25	45	45	65	43, 36, 49, 33, 12, 16, 10, 58, 2, 15
6	6, 26, 32, 9, 22, 18, 17, 57, 0, 43, 1	26	26	46	46	66	48, 58, 41, 56, 21, 27, 6, 37, 4, 55
7	7, 24, 44, 5, 45, 22, 19, 30, 46 15, 9	27	27	47	47	67	57, 44, 24, 26, 3, 0, 19, 55, 7, 33

8	8, 3, 44, 52, 13, 12, 59, 26, 18, 60, 28	28	28	48	48	68	12, 43, 11, 29, 6, 13, 28, 53, 58, 50
9	9	29	29	49	49	69	29, 49, 41, 8, 48, 40, 44, 36, 3, 47
10	10	30	30	50	50	70	48, 41, 43, 9, 16, 56, 49, 53, 17, 57
11	11	31	31	51	51	71	11, 46, 43, 15, 1, 9, 12, 55, 59, 31
12	12	32	32	52	52	72	49, 42, 57, 11, 43, 0, 21, 20, 17, 2
13	13	33	33	53	53	73	41, 29, 49, 44, 10, 60, 39, 8, 47, 57
14	14	34	34	54	54	74	12, 47, 23, 55, 60, 57, 46, 56, 13, 7
15	15	35	35	55	55	75	49, 8, 46, 53, 0, 9, 11, 58, 56, 50
16	16	36	36	56	56	76	30, 48, 57, 51, 49, 12, 15, 44, 52, 56
17	17	37	37	57	57	77	51, 12, 4, 48, 11, 30, 55, 13, 34, 49
18	18	38	38	58	58	78	27, 8, 46, 15, 7, 59, 44, 5, 55, 1
19	19	39	39	59	59		
20	20	40	40	60	60		

Note: ID: Presenter's and participant's id number, Paper's ID: Presentation's id number.

The conference has 60 papers to be presented from 60 different presenters, and there were 18 participants who were not presenting. Thus, the total number of data is 78. Table 1 shows that there were 8 presenters (ID 1 until 60 is the presenters ID) and 18 participant (ID 61 until 78 is the participants ID) that have chosen their preferences. We can see that not everyone that answer the questionnaire gave all 10 choices of papers that they were interested. Some only gave 9 preferences of papers instead of 10.

5.2.Experimental Results

As mentioned earlier, there are several steps involve in the DTA process where the first step is to obtain the information from the participants-papers data file. This step is more into managing and organizing the dataset into the standard form that we preferred as shown in Table 1. After sorting out the data, step 2 is to do the pre-processing. In this pre-processing step, from the standard form of data that we preferred, we process the data to construct the conflict matrix.

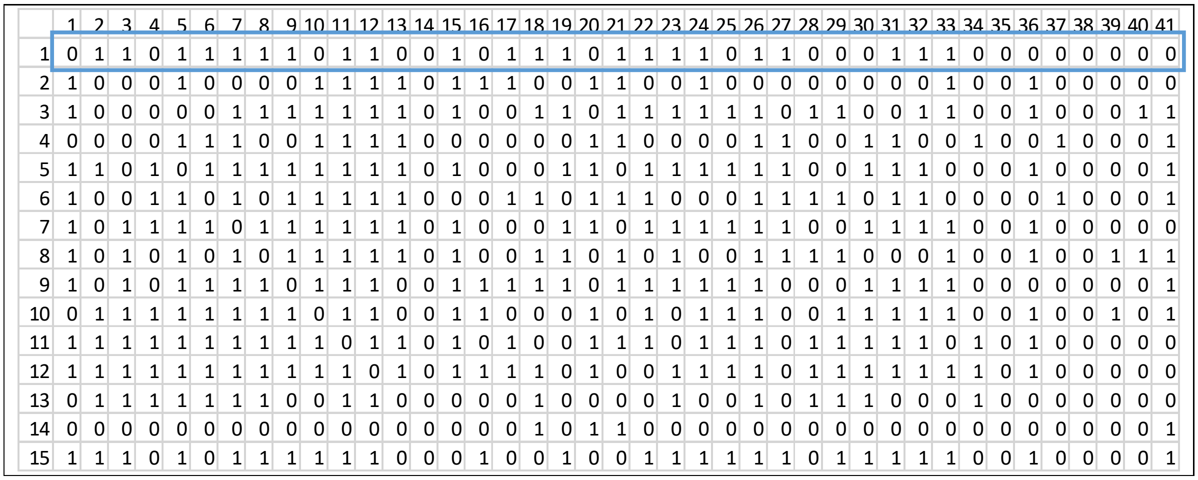
The actual dimension of the conflict matrix for this conference is 60 by 60. Fig. 3 shows the fraction (segmentation of 41 by 15 matrix cropped from the 60 by 60 matrix, due to limited space in this paper) of the conflict matrix for SKSK06. The value of 0 (zero) in the matrix at location (i,j) indicates that paper i is not in conflict with paper j. On the other hand, the value of 1 (one) indicates that those two papers are in conflict with each other. For example, the value at the intersection of row 1 and column 13 is 0. That means that paper 13 is not conflicting with paper 1. Based on this understanding, from the conflict matrix, we can obtain the information on a list of conflicting papers and non-conflicting papers for paper 1 as follows:

1) Conflicting papers (value 1):

Paper 1: 2 3 5 6 7 8 9 11 12 15 17 18 19 21 22 23 24 26 27 31 32 33 42 43 44 46 47 49 51 55 57 59 60

2) Non-conflicting papers (value 0):

Paper 1: 4 10 13 14 16 20 25 28 29 30 34 35 36 37 38 39 40 41 45 48 50 52 53 54 56 58



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41
1	0	1	1	0	1	1	1	1	1	0	1	1	0	0	1	0	1	1	1	0	1	1	1	1	0	1	1	0	0	0	1	1	1	0	0	0	0	0	0	0	0
2	1	0	0	0	1	0	0	0	0	1	1	1	1	0	1	1	1	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0
3	1	0	0	0	0	0	1	1	1	1	1	1	1	0	1	0	0	1	1	0	1	1	1	1	1	1	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1
4	0	0	0	0	1	1	1	0	0	1	1	1	1	0	0	0	0	0	0	1	1	0	0	0	0	1	1	0	0	1	1	0	0	1	0	0	1	0	0	0	1
5	1	1	0	1	0	1	1	1	1	1	1	1	1	0	1	0	0	0	1	1	0	1	1	1	1	1	1	0	0	1	1	1	0	0	0	1	0	0	0	0	1
6	1	0	0	1	1	0	1	0	1	1	1	1	1	0	0	0	1	1	0	1	1	1	0	0	0	1	1	1	1	0	1	1	0	0	0	0	1	0	0	0	1
7	1	0	1	1	1	1	0	1	1	1	1	1	1	0	1	0	0	0	1	1	0	1	1	1	1	1	1	0	0	1	1	1	1	0	0	1	0	0	0	0	0
8	1	0	1	0	1	0	1	0	1	1	1	1	1	0	1	0	0	1	1	0	1	0	1	0	0	1	1	1	1	0	0	0	1	0	0	1	0	0	1	1	1
9	1	0	1	0	1	1	1	1	0	1	1	1	0	0	1	1	1	1	1	0	1	1	1	1	1	1	0	0	0	1	1	1	1	0	0	0	0	0	0	0	1
10	0	1	1	1	1	1	1	1	1	0	1	1	0	0	1	1	0	0	0	1	0	1	0	1	1	1	0	0	1	1	1	1	1	0	0	1	0	0	1	0	1
11	1	1	1	1	1	1	1	1	1	1	0	1	1	0	1	0	1	0	0	1	1	1	0	1	1	1	0	1	1	1	1	1	0	1	0	1	0	0	0	0	0
12	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1	0	1	0	0	1	1	1	1	0	1	1	1	1	1	1	1	0	1	0	0	0	0	0
13	0	1	1	1	1	1	1	1	0	0	1	1	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0	1	1	1	0	0	0	1	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
15	1	1	1	0	1	0	1	1	1	1	1	1	0	0	0	1	0	0	1	0	0	1	1	1	1	1	1	0	1	1	1	1	1	0	0	1	0	0	0	0	1

Fig.3. Fraction of the conflict matrix

Next step is the scheduling process. This is where the papers will be assigned to each time slot, while generating the conflict chain as were explained earlier in the methodology section. Conflict chain as mentioned earlier is a data representation that one can use to determine the least number of time slot needed in generating a feasible schedule. Table 2 shows the result obtained after applying the conflict chain algorithm. Based on the result, if we were to check with the information from the conflict matrix, we can see that the result is a feasible schedule where each paper in the same time slot is not conflicting with each other based on the preferences made by the presenters and participants before. The best number of time slot that we can generate in this schedule is 15 time slots in spite of the fact that we cater all the preferences made by the participants. This result is consistent with the result derived from the research made in [21], which means a feasible schedule has been produced which satisfy both of the constraints in terms of scheduling all papers with the same presenters without conflict and so as scheduling all papers by satisfying all participants' preferences without any conflicts.

Table 2. Result obtained from the experiment

Time Slot	Papers Scheduled (Based on Paper's ID)						
1	12	22	35	38	41		
2	24	49					
3	4	25	40	45	50	57	59
4	11	16	18	27	47		
5	9	13	20	36	39		
6	6	14	15	34	60		
7	10	42	52	53	54	55	
8	1	28	48				
9	3	5	56				
10	8	37	43	51			
11	17	23	44	58			
12	2	7	29				
13	19	31	32				
14	21	26	30				
15	33	46					

Apart from that, another hard constraint in this study is to overcome the capacitated constraint. Table 2 might have the feasible schedule but the number of papers in each time slot is not balance or even. Some have 2 papers in a time slot, some have 3 papers and there is also a time slot with 7 papers in it. It is inappropriate to have a schedule that have imbalance paper per time slot because every time slot has the same duration of time provided. For example, if a time slot has 2 hours' duration of time, it is hardly possible to fit in 7 papers (with an average of about 20 to 30 minutes per presentation as per normal conference practise) that need to be presented between the periods of time given.

Also, in another perspective of participations satisfactions, it might be not fair for some presenters to have a long queue to wait to present their paper, if they were scheduled to a late session in a time slot with more papers (for example a slot with 7 paper) as opposed to presenters scheduled to a slot with a fewer papers (for example a slot with 2 papers). Thus, the last process in DTA which is the optimization process will be performed to generate a better quality schedule with a more balanced number of paper in a time slot.

Table 3. Result obtained from the optimization process (maximum of 6 papers per slot)

Time	Papers Scheduled					
Slot	**(Based on Paper's ID)**					
1	12	22	35	38	41	
2	24	49	59			
3	4	25	40	45	50	57
4	11	16	18	27	47	
5	9	13	20	36	39	
6	6	14	15	34	60	
7	10	42	52	53	54	55
8	1	28	48			
9	3	5	56			
10	8	37	43	51		
11	17	23	44	58		
12	2	7	29			
13	19	31	32	40		
14	21	26	30			
15	33	46				

Table 4. Result obtained from the optimization process (maximum of 5 papers per slot)

Time	Papers Scheduled				
Slot	**(Based on Paper's ID)**				
1	12	22	35	38	41
2	24	49	59	54	
3	4	25	40	45	57
4	11	16	18	27	47
5	9	13	20	36	39
6	6	14	15	34	60
7	10	42	52	53	55
8	1	28	48		
9	3	5	56		
10	8	37	43	51	
11	17	23	44	58	
12	2	7	29		
13	19	31	32	50	
14	21	26	30		
15	33	46			

Meanwhile, Table 4 portrayed the result with the maximum number of 5 papers per slot. From the result, we can see that 2 papers from time slot 3 and a paper from time slot 7 is effected by this optimization process. After undergoing the optimization process, time slot 2 now has 4 papers as compared to before which is only 2 papers. As for time slot 13, 1 additional paper has been added to it. Even though time slot 15 has the least number of papers based on the optimization algorithm, if a particular paper is conflicting with each paper in the slot that has least number of papers, we need to find another paper from the time slot having number of papers more than the limit allowed. Otherwise, find other slot that can first fit in the paper into the slot without any conflicts.

Table 3 consists of the result after performing optimization to balance out the number of papers in the time slots. For this case, we aim at setting a limit of a maximum number of 6 papers per slot. During the optimization, after traversing, paper with ID of 59 in time slot 3 is identified as the paper that needs to be moved since it is the 7th paper in the slot.

Thus, the optimization procedure placed this exceeding paper into time slot 2 as it was the first slot identified that can be used to assign this particular paper without any conflicts.

Table 5. Result obtained from the optimization process (maximum of 4 papers per slot)

Time Slot	Papers Scheduled(Based on Paper's ID)			
1	12	22	35	41
2	24	49	38	59
3	4	25	40	57
4	11	16	27	47
5	9	13	20	36
6	6	14	15	60
7	10	42	54	55
8	1	28	48	45
9	3	5	56	34
10	8	37	43	51
11	17	23	44	58
12	2	7	29	18
13	19	31	32	50
14	21	26	30	53
15	33	46	39	52

Table 5 depicted the result obtained from the experiment when only a maximum of 4 papers is allowed in a slot. It is seen that in this study, this is the most optimum schedule generate among others, as it balanced out the number of papers in each time slot where all slots now are having 4 papers each. This feasible schedule generated can be considered a very good quality schedule since all preferences made by the participants were managed to be satisfied. In brief, it is obvious that the constructed schedule generated using DTA is of high quality where all the papers were scheduled efficiently without any clashes in just 15 time slots by satisfying all the preferences made. Moreover, the number of papers assigned to time slots is balanced as aimed in this study.

6. CONCLUSION

Concisely, this study is re-applying the DTA that were originally proposed for examination scheduling problem, to solve the SKSK06 conference dataset that is considered as preference-based conference scheduling problem. A capacitative constraint was added to the original problem to balance out the number of papers in each time slot. This study has 3 main objectives to achieve, which is to generate a feasible schedule, to maximize the satisfactions of participants by increasing the number of presentations that the participants want to attend and lastly is to balance out the number of papers in a time slot. Based on the results obtained, we can perceive that DTA is successfully generating not only a feasible schedule based on participants' preferences as well as even out the number of papers per slot.

Despite the encouraging results produced, we believed that the schedule can still be improved. For example, producing schedules with parallel sessions so it will only takes less number of days to run the conference. Other possible future work might be grouping the papers according to the same keywords so that we can minimize session hopping. Thus, participants or presenters do not need to move from one room to another for attending sessions of interests.

Lastly, we conclude that Domain Transformation Approach (DTA) is flexible not only for Examination Scheduling Problem (ESP)[4, 13], it is also applicable to conference scheduling problem and perhaps any other scheduling problem. Through DTA, scheduling process has become easier and simpler and managed to generate competitive schedules.

7. ACKNOWLEDGEMENTS

This work is financially supported by the Ministry of Education Malaysia and UniversitiTeknologi MARA, Malaysia under FRGS grant scheme with number 600-RMI/FRGS 5/3 (129/2015).

8. REFERENCES

[1] Abdul R S K N, Bargiela A, Qu R. Granular modelling of exam to slot allocation.In European Conference on Modelling and Simulation, 2009, pp. 861-866

[2] Abdul R S K N, Bargiela A, Qu R. Domain transformation approach to deterministic optimization of examination timetables. Artificial Intelligence Research, 2012, 2(1):122-138

[3] Abdul R S K N, Bargiela A, Qu R. The incorporation of late acceptance hill climbing strategy in the deterministic optimization of examination scheduling framework: A comparison with the traditional hill climbing.In IEEE Conference on Systems, Process and Control,2014, pp. 147-152

[4] Abdul R S K N, Jaafar A H, Bargiela A, Zulkipli F. Solving the preference-based conference scheduling problem through domain transformation approach. In International Conference on Computing and Informatic, 2017, pp. 199-207

[5] Akbulut A, Yılmaz G. University exam scheduling system using graph coloring algorithm and RFID technology.International Journal of Innovation, Management and Technology, 2013, 4(1):66-72

[6] Aycan E, Ayav T. Solving the course scheduling problem using simulated annealing.In IEEE InternationalAdvance Computing Conference, 2009, pp. 462-466

[7] Baskaran G. A domain transformation approach for addressing staff scheduling problems.Phd thesis, England: University of Nottingham, 2016

[8] Baskaran G, Bargiela A, Qu R. A study of cost effective scheduling of nurses based on the domain transformation method. In 27th European Conference on Modelling and Simulation, 2013, pp. 309-314

[9] Baskaran G, Bargiela A, Qu R. Simulation of scheduling and cost effectiveness of nurses using domain transformation method. In European Conference on Modelling and Simulation, 2014, pp. 226-234

[10] Baskaran G, Bargiela A, Qu R. Domain transformation using greedy algorithm in nurse scheduling. In International Conference on Artificial Intelligence, Energy and Manufacturing Engineering, 2015, pp. 92-98

[11] Bellardo H. Preference driven university course scheduling system.Master thesis, San Luis Obispo: California Polytechnic State University, 2010

[12] Carrington L G. The impact of course scheduling on student success in intermediate accounting.American Journal of Business Education, 2010, 3(4):51-60

[13] Edis E B, Edis RS. An integer programming model for the conference timetabling problem.C.B.U. Journal of Science, 2013, 9(2):55-62

[14] Eley M. Ant algorithms for the exam timetabling problem. In International Conference on the Practice and Theory of Timetabling, 2006, pp. 167-180

[15] Gulati M, Sengupta A. TRACS-Tractable conference scheduling. In Proceedings from Decision Science Institute Annual Meeting, 2004, pp. 3161-3166

[16] Ibrahim H, Ramli R, Hassan M H. Combinatorial design for a conference: Constructing a balanced three-parallel session schedule. Journal of Discrete Mathematical Sciences and Cryptography, 2008,11(3):305-317

[17] Komijan A R, Koupaei M N. A mathematical model for university course scheduling: A case study.International Journal of Technical Research and Applications, 2015, 3(19):20-25

[18] Malkawi M, Hassan M A, Hassan O A. A new exam scheduling algorithm using graph coloring.International Arab Journal of Information Technology, 2008, 5(1):80-87

[19] Ojeyinka T O. Bin packing algorithms with applications to passenger bus loading and multiprocessor scheduling problems.Communications on Applied Electronics, 2015, 2(8):38-44

[20] Ozcan S O, Dokeroglu T, Cosar A, Yazici A. A novel grouping genetic algorithm for the one-dimensional bin packing problem on GPU. In International Symposium on Computer and Information Sciences, 2016, pp. 52-60

[21] Quesnelle J, Steffy D. Scheduling a conference to minimize RSVP conflicts. 2015, http://jeffq.com/blog/wp-content/uploads/2015/04/penguicon_schedule_preprint.pdf

[22] Renault M P, Rosén A, Stee R Van. Online algorithms with advice for bin packing and scheduling problems. Theoretical Computer Science, 2015, 600:155-170

[23] Sampson S E. Practical implications of preference-based conference scheduling. Production and Operations Management, 2004, 13(3):205-215

[24] Sampson S E, Weiss E N. Designing conferences to improve resource utilization and participant satisfaction. Journal of the Operational Research Society, 1996, 47(2):297-314

[25] Stee R V. On-line scheduling and bin packing. Phd thesis, Leiden: University of Leiden,

2002

[26] Zulkipli F, Ibrahim H, Benjamin A M. Optimization capacity planning problem on conference scheduling. In IEEE Business Engineering and Industrial Applications Colloquium, 2013, pp. 911-915

[27] Coffman E G, Leung J Y T, Ting D W. Bin packing: Maximizing the number of pieces packed. ActaInformatica, 1978, 9(3):263-271

[28] University of Nebraska-Lincoln. Summary of bin-packing algorithms. 2006, http://www.math.unl.edu/~s-sjessie1/203Handouts/Bin%20Packing.pdf

[29] Stee R V.Bin packing and scheduling. 2010, https://i11www.iti.kit.edu/_media/teaching/sommer2010/approximationsonlinealgorithmen/bp-sched.pdf

How to cite this article:
Jaafar S N, Abd Rahim S K N, Omar N, Masrom S, and Khairudin N.Incorporating Capacitative Constraint to Preference-Based Conference Scheduling via Domain Transformation Approach. J. Fundam. Appl. Sci., 2017, *9(5S), 404-422.*