Enhancing the Sorting Layers in the Initial Stage of High School Timetabling

Billel Arbaoui
School of Computing,
Universiti Utara Malaysia
Sintok, Kedah, Malaysia
billel_arbaoui@ahsgs.uum.edu.my

Juliana Wahid
School of Computing,
Universiti Utara Malaysia
Sintok, Kedah, Malaysia
w.juliana@uum.edu.my

Syariza Abdul-Rahman
School of Quantitative Sciences,
Universiti Utara Malaysia
Sintok, Kedah, Malaysia
syariza@uum.edu.my

***Abstract*—The high school timetabling problem (HSTP) is considered as an NP-Complete problem as the optimal solution for it, is still not discovered by any algorithm. Generally, NP-Complete problem was solved firstly by constructing the initial solution, in the construction phase. The initial solution will be improvised in the improvisation phase. KHE is an algorithm that generates initial solution of HSTP. The layer sorting procedure in KHE is based on a certain priority. For every two layers, the layers will be ranked based on the highest value of duration. If both layers have equal value of duration, the layer with the highest value of demand will be at a higher rank. If both layers have equal value of demand. The layer will be arranged according to the index value of the layer. These sorting criteria use the layer properties independently which causes non-good results after the time-assignment phase. Therefore, this study proposed a mathematical model based on the Markov Chain Model for the sorting procedure that combines the layer properties in a formula. The proposed model was executed with 40 datasets of XHSTT2014, and it shows better results on 25 datasets of XHSTT2014 compared to the KHE algorithm. The mathematical model based on Markov Chain proposed in this study is able to improvise the original sorting of KHE.**

Keywords*—High School Timetabling, XHSTT, KHE, Construction Algorithm, Markov Chain

I. INTRODUCTION

The high school and university scheduling for courses and examination is typical in educational institution timetabling [1], [2]. The high school timetable is weekly-based for all classes, avoiding conflict or overlap of teacher and time slot [3], [4]. The research in school timetabling problem (STP) has started since 1975 [5] and is still extremely active. The solution for STP comprises of numerous algorithms and methods include *GRASP* algorithm with *Tabu Search* algorithm [6], hybrid local search [7], hyper-heuristics [8], and parallel local search [9]. The STP is classified as an NP-complete problem, which means that the current algorithms and methods still unable to find an optimal solution for the problem [10].

An algorithm for solving high school timetabling named as Kingston High School Timetabling Engine (KHE) which produced by Kingston [11] schedules meetings, times, and resources for particular term or semester. KHE was produced based on several works of Kingston on STP structures from 2005 to 2014 [11]–[14]. The dataset used by KHE is known as XML High School Timetabling (XHSTT) that represent real instances and solutions of the high school problems. The XHSTT which contributed by several countries such as Australia, Brazil, Italy, USA etc. were also used as benchmark dataset in the Third International Timetabling Competition 2012 [15], [16]. KHE is widely used by researchers in order to produce the STP solution [7]–[9], [17], [18].

The structure of KHE [11] algorithm consists of four phases which are the structure phase, times assignment phase, resources assignment phase and cleanup phase as shown in Fig. 1. The structure phase will arrange the events without the times and resources assignment based on ten constraints (split event, distribute split events, spread events, avoid split assignment, link event, preferred time, preferred resource, unavailable time, limit busy time and limit workload constraints). The output of this phase is the STP solution (without the times and resources), and is produced in a tree of layers that have nodes inside it. Each node contains several events. The second phase is the times assignment phase. This phase will assign the time into the STP solution produced in the structure phase based on five constraints. The first constraint is a hard avoid clashes constraint that applied for modifying the layers produced in the structure phase. The other constraints are assign times, cluster busy times, limit idle times, and limit busy times constraints. The output of this phase is the STP solution with the times but without the resources. The third phase is the resources assignment phase. The resources consist of teachers, classes, rooms and students. This phase will assign the resources into STP solution based on two constraints which are avoid split assignment and assign resources constraints. This phase is carried out in sequence, where the teachers will be assigned first and followed by the classes, rooms and students. The output of this phase is the STP solution that has the events with the times and the resources assigned to it. The final phase is the cleanup phase that removes the duplication of events that have the same times and resources assignment in the STP solution.

The layers that were produced in the structure phase will be sorted in the time assignment phase based on certain criteria. The sorting procedure will use four conditions which are 1) if the layer is visited, the non-visited layer should be sorted first, 2) if both layers are non-visited, the layer with the highest number of durations will be sorted first, 3) if both layers have an equal number of durations, the layer with the highest number of demands will be sorted first, 4) if both layers have an equal number of demands and durations, the sorting will be based on the index/position number of the layers [19].

The sorting procedures focusing on the layer properties (duration, demand and index) individually. Therefore, this study proposes a sorting procedure that combine all the layer properties in a mathematical formula based on Markov Chain model.

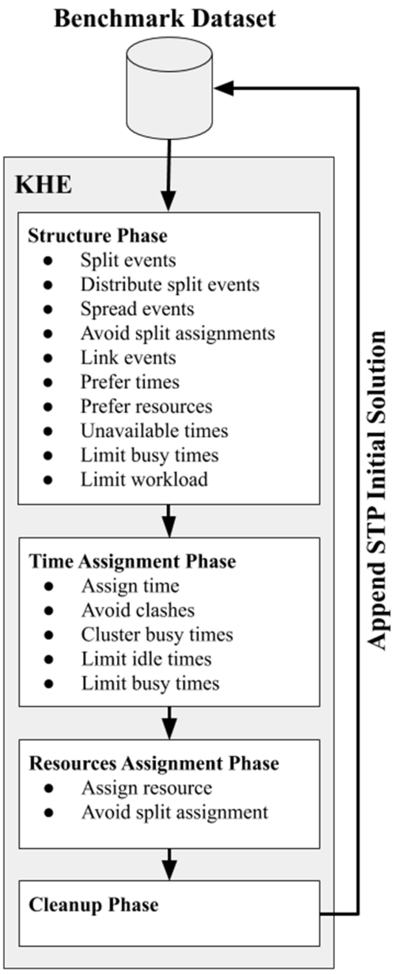

Fig. 1. Major Steps of KHE

The rest of the paper is organized as followed. The second section describes the XML high school timetabling (XHSTT). The proposed model is described in section three. Later, the experiments and results are presented in section four. Finally, the limitations and future research, and conclusions are mentioned in section five.

II. THE XML OF HIGH SCHOOL TIMETABLING

The XML high school timetabling (XHSTT) model represents the high school timetabling problem. The XHSTT is mapped out by XML schema that describes the real-world problem data of STP. The objective of XHSTT is to cover the features of STP. The instances of XHSTT have four tags which are:

- **XHSTT Times**: a set of time slot in a day.
- **XHSTT Resources**: consists of teachers, classes, rooms or students. There is also a set of resources for a group of specific teachers (such as mathematic or history).
- **XHSTT Events**: is a meeting between resources and times that gives information such as duration, workload, time and resources. Duration represents the timeslots number which have to be assigned to the event, and some events have pre-assigned timeslots [20]. Workload is number of events that related to specific resources and duration. There is a set of events for specific feature grouped events such as sport or lab resources.
- **XHSTT Constraints:** defines the conditions on the instances. There are 16 constraints as listed in Table 1. Each constraint gives a set of points application which has relationship on how the cost will be calculated and where constraint (at that point) fails to satisfy its condition. The cost is calculated by the following Equation 1:

$$Cost = Weight * CostFunction(deviation) \quad (1)$$

Where weight is determined by XHSTT constraints in a range between 0 to 1000, and cost function will calculate the deviation. There are three types of cost functions (deviation): Linear, Quadratic, and Step. Also, each constraint can be categorized as soft or hard based on the XHSTT instances.

TABLE 1. THE CONSTRAINTS IN THE XHSTT FORMAT

	Name	Description
1	Assign Resource	specify the solution events should be assigned with the resources
2	Assign Time	specify the solution events should be assigned with the times
3	Split Events	set limits on the number of solution events based on the instance event, and on its durations
4	Distribute Split Events	put limits on the number of solution events of a specific duration based on the instance event
5	Prefer Resources	state that some “resources” are preferable for assignment to some solution resources (in solution events)
6	Prefer Times	state that some “times” are preferable for assignment to some solution events
7	Avoid Split Assignments	identify the split assignments are unfavorable for some event resources
8	Spread Events	list the solution events of an event group should be spread out based on time
9	Link Events	specify some events should be assigned with the same times
10	Order Events	specify the times of two events should be in order
11	Avoid Clashes	set certain resources should have no clashes
12	Avoid Unavailable Times	specifies some resources are unavailable to attend any events at certain times
13	Limit Idle Times	set limits on the number of times that resources may be idle
14	Cluster Busy Times	identify limits on the number of time groups during which a resource may be busy
15	Limit Busy Times	list limits on the number of times during certain time groups that a resource may be busy
16	Limit Workload	set limits on the total workload (minimum and maximum) of solution resources (in solution events) that certain resources are assigned to it

III. THE PROPOSED MODEL

Markov Chain is a mathematical model that transits from one state to another which used to connect the ideas/state/something according to certain probabilistic rules [21], [22]. In this paper, a static Markov Chain is applied to calculate the value of certain layer based on the number of durations (*Du*), demands (*De*), and index (*Ix*) of the layers. Duration represents the timeslots number which have to be assigned to the event [20]. Demand is a relation between meets and times. Meets are the events or sub-events. Index is the position of layer.

The conceptual of sorting layers as shown in Fig. 2 consists of *Du*, *De*, and *Ix* which have the weightage that based on the priority of the concepts. Using α which represents the weight of each concept, *Ix* has $1-\alpha$ as it has the lowest priority affected to *De*. The equation of the *Ix* is as follow:

$$Ix = (1-\alpha) \,.\, Ix \qquad (2)$$

In the other hand, *De* has the weight of α affected to itself with the higher priority compared to *Ix*. The equation of *De* is shown below:

$$De = \alpha \,.\, De + Ix \qquad (3)$$

Du has the highest priority compared to *De* and *Ix*. *Du* has the weight α in producing the layer value. The equation of the layer value which also includes the Equation 3 is shown below:

$$Value = \alpha \,.\, Du + (1-\alpha) \,.\, De \qquad (4)$$

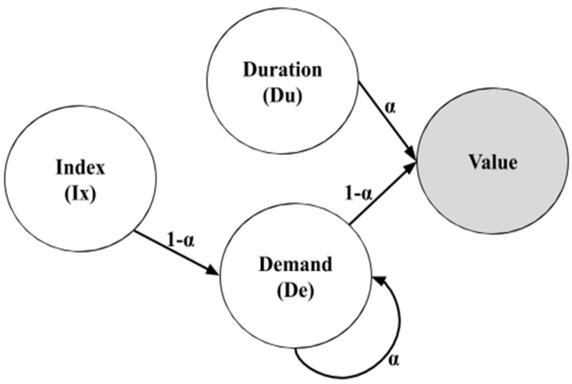

Fig. 2. Conceptual of Sorting Layers

As in Markov Chain model, the value used only between zero and one [0.1 - 0.9], the value of *Du* and *De* in the Equation 4 needs to be normalized as 0.1 to 0.9 using the formula:

$$\text{Normalized Value} = \frac{Value\ of Du/De - Minimum(Du/De)}{Maximum(Du/De) - Minimum(Du/De)} \qquad (5)$$

For example, the value of *Du* for layer one is 30, the maximum value of *Du* in other layers is 98, and minimum value of Du in other layer is 5. So, the normalized value for *Du* is:

$$Normalized\ Value = \frac{30-5}{98-5} \approx 0.27$$

The layer comparison procedure as in Equation 4 is shown in Fig. 3. The flowchart of the comparison procedure has two conditions which are: 1) if one of the layer is already assigned a time to it, the other layer will be sequenced to the top, 2) if both layers are not assigned time to it, the Equation 4 will be applied to calculate value of both layers. The higher value of both layers will be sequenced to the top.

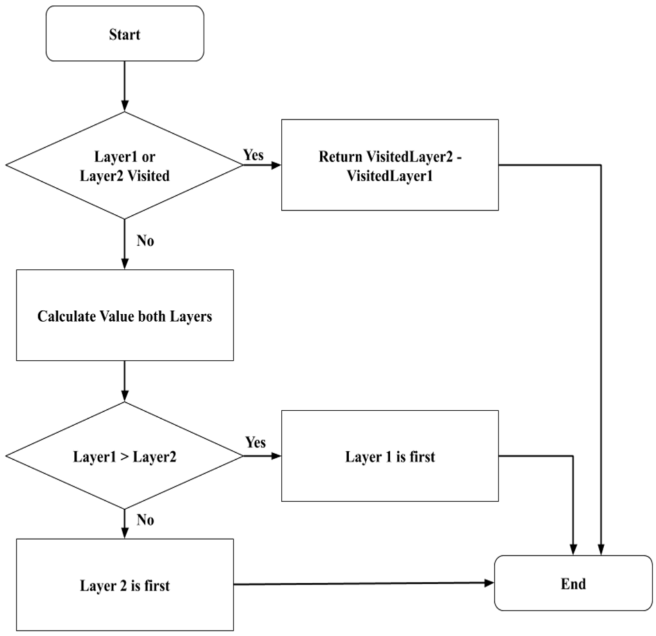

Fig. 3. Flowchart of The Comparison Procedure for Sorting Layers during Time Assignment Phase

IV. EXPERIMENT AND RESULTS

Based on the preliminary experiments, the value of α is set to 0.99 as this value has produced a good result in the total penalty cost. The proposed model discussed in Section III was developed in the KHE algorithm by using C language with an Intel Core i7 with 3.60 GHz processor and 16GB RAM. In order to evaluate the proposed model, different XHSTT datasets (real problem of high school timetabling) from small to huge instances [16] were used. The summary of the XHSTT datasets as shown in Table 2 consists of 40 datasets from 13 countries that comprises of properties such as instance names, times, resources (teachers, rooms, students and classes), events, and total number of duration. The dataset from *Czech* named as *CzechVillageSchool* is considered as a small dataset as the total number of duration is 68, which is the lowest total number of duration compared to other datasets, while the dataset from *USA* named as *US-WS-09* is a huge dataset with 6354 of total number of duration.

Each dataset in Table 2 was used in the execution of the proposed model according to the number of threads implemented in KHE16. Table 3 shows the results of the proposed model compared with the results of KHE16 [23] for each dataset. The result contains the total number of hard constraints violations and the total number of soft constraints violations. For example, a value of 3.00608 denotes that the total number of hard constraint violations is 3 and the total penalty value of soft constraints violations is 608. In [23], the solution of *VillageSchool* dataset was not presented and another 14 datasets. This study executed the KHE16 for this dataset in order to make its solution available for comparison.

TABLE 2. A SUMMARY OF THE XHSTT2014 INSTANCES

	Instance	Times	Resources	Events	Duration
Australia	AU-BG-98	40	131	387	1564
	AU-SA-96	60	99	296	1876
	AU-TE-99	30	76	308	806
Brazil	BR-SA-00	25	20	63	150
	BR-SM-00	25	35	127	300
	BR-SN-00	25	44	140	350
	BrazilInstance5	25	44	119	325
	BrazilInstance3	25	24	69	200
	BrazilInstance7	25	53	205	500
	BrazilInstance1	25	24	21	75
Czech	VillageSchool	30	14	67	68
Denmark	DK-FG-12	50	438	1077	1077
	DK-HG-12	50	694	1235	1235
	DK-VG-09	60	262	918	918
Spain	ES-SS-08	35	91	225	439
Finland	FI-PB-98	35	111	387	854
	FI-WP-06	35	41	172	297
	FI-MP-06	35	64	280	306
	SecondarySchool2	40	79	469	566
	ElementarySchool	35	103	291	445
	ArtificialSchool	20	47	169	200
Greece	GR-H1-97	35	95	372	372
	GR-P3-10	35	114	178	340
	GR-PA-08	35	31	262	262
	UniversityInstance3	35	25	210	210
	UniversityInstance5	35	24	184	184
	Preveza2008	35	98	164	340
	Aigio2010	35	245	283	532
Italy	IT-I4-96	36	99	748	1101
	ItalyInstance1	36	16	42	133
Kosovo	KS-PR-11	62	164	809	1912
Netherlands	NL-KP-03	38	587	1156	1203
	NL-KP-05	37	644	1235	1272
	NL-KP-09	38	194	1148	1274
	Kottenpark2008	40	126	1027	1095
	GEPRO	44	1102	2675	2675
UK	UK-SP-06	25	202	1227	1227
USA	US-WS-09	100	242	628	6354
South Africa	ZL-LW-09	148	37	185	838
	ZA-WD-09	42	70	278	1253

In Table 3, the proposed model is able to enhance the solutions produced by the KHE16 in 25 out of 40 instances. The solutions of these 25 instances are in bold. The proposed model performs the best on *AU-SA-96* and *AU-TE-99* by satisfying all the hard constraints and minimizing the soft constraints violations of the KHE16. However, even the proposed model can minimize the soft constraints violations in *AU-BG-98* and *NL-KP-05*, but the hard constraints for these datasets were slightly increased. The reason of decreasing or increasing hard/soft constraints based on the constraints of dataset, because of that proposal model is static where the equations are work with non-dynamically model. There are four tied solutions in italic which are *GR-H1-97, GR-P3-10, GR-PA-08* and *UniversityInstance5*.

TABLE 3. RESULTS OBTAINED USING PROPOSED MODEL COMPARED TO KHE16

	Instance	KHE16	Proposed Model
Australia	AU-BG-98	**3.00608**	4.00561
	AU-SA-96	4.00022	**0.00006**
	AU-TE-99	2.00152	**0.00087**
Brazil	BR-SA-00	0.00031	**0.00011**
	BR-SM-00	8.00177	**2.00125**
	BR-SN-00	0.00113	**0.00090**
	BrazilInstance5	0.00103	**0.00083**
	BrazilInstance3	**0.00062**	0.00065
	BrazilInstance7	0.00135	**0.00134**
	BrazilInstance1	**0.00041**	0.00048
Czech	VillageSchool	0.00014	**0.00013**
Denmark	DK-FG-12	0.03411	**0.03319**
	DK-HG-12	**12.04689**	12.04702
	DK-VG-09	2.04691	**2.04184**
Spain	ES-SS-08	0.00657	**0.00565**
Finland	FI-PB-98	**0.00000**	0.00020
	FI-WP-06	**0.00019**	0.00024
	FI-MP-06	**0.00102**	0.00105
	SecondarySchool2	0.00010	**0.00005**
	ElementarySchool	**0.00003**	0.00003
	ArtificialSchool	9.00011	**8.00003**
Greece	GR-H1-97	*0.00000*	*0.00000*
	GR-P3-10	*0.00002*	*0.00002*
	GR-PA-08	*0.00011*	*0.00011*
	UniversityInstance3	0.00007	**0.00006**
	UniversityInstance5	*0.00000*	*0.00000*
	Preveza2008	0.00009	**0.00002**
	Aigio2010	**0.00014**	0.00015
Italy	IT-I4-96	0.00046	**0.00037**
	ItalyInstance1	0.00024	**0.00022**
Kosovo	KS-PR-11	0.00017	**0.00012**
Netherlands	NL-KP-03	0.01371	**0.00135**
	NL-KP-05	**15.10117**	19.05859
	NL-KP-09	10.05125	**7.04190**
	Kottenpark2008	24.40431	**19.35165**
	GEPRO	1.01866	**1.01847**
UK	UK-SP-06	45.01304	**9.00836**
USA	US-WS-09	**0.00532**	0.00537
South Africa	ZL-LW-09	13.00016	**2.00034**
	ZA-WD-09	16.00000	**2.00000**

Next, the comparison between KHE16 and the proposed model in terms of duration and demand values obtained of each layer is presented in Table 4. The table shows the analysis on "*AU-TE-99*" after time assignments phase. And, the KHE16 gives the position of layer two, (the value of duration and demands in bold), that should be the latest one to assign the time, while the proposed model calculates the position that the layer should be the latest one, and thus the violated constraints is less than KHE16.

TABLE 4. COMPARISON OF ORDER LAYERS BETWEEN KHE16 AND THE PROPOSED MODEL FOR "*AU-TE-99*"

	After Time Assignments Phase			
	KHE16		Proposal Model	
Layers	Du	De	Du	De
Layer 1	30	224	30	302
Layer 2	**6**	**18**	30	224
Layer 3	14	163	30	224
	•	•	•	•
	•	•	•	•
	•	•	•	•
	•	•	•	•
Layer N	17	127	**6**	**18**

V. CONCLUSIONS

This study proposes a sorting procedure that involved a mathematical formula based on the Markov Chain model in the time assignment phase of KHE. The proposed model is able to produce better STP solution result on certain XHSTT dataset compared to KHE. The result of this study can be used in the improvement phase in solving the STP.

The proposed model is a static model that tries to show the connection between three conceptual states during the time assignment phase. This proposed model uses simple equation which studies the three states in non-dynamic model (e.g. without the sequence time). From the results, some XHSTT instances could not be enhanced in terms of their solution compared to KHE. From this point of view, the dynamic model based on differential equations perhaps would produce better initial solutions for all XHSTT datasets. Other future work of this study is to come out with combination approaches using meta-heuristics based population to improve the STP initial solution.

ACKNOWLEDGEMENTS

The authors want to thank the Ministry of Higher Education for funding this study under the Fundamental Research Grant Scheme, S/O code 13796 and RIMC, Universiti Utara Malaysia for the administration of this study.

REFERENCES

[1] N. Pillay, "A survey of school timetabling research," *Ann. Oper. Res.*, vol. 218, no. 1, pp. 261–293, 2014.

[2] J. Wahid, S. Abdul-Rahman, A. M. Din, S. M. Yusof, and N. M. Hussain, "Solving Curriculum Based University Course Timetabling Problem: Trends on Metaheuristic Approaches," *ICT Transform.*, p. 42, 2018.

[3] A. Schaerf, "Survey of automated timetabling," *Artif. Intell. Rev.*, vol. 13, no. 2, pp. 87–127, 1999.

[4] S. Faudzi, S. Abdul-Rahman, and R. Abd Rahman, "An Assignment Problem and Its Application in Education Domain: A Review and Potential Path," *Adv. Oper. Res.*, vol. 2018, 2018.

[5] S. Even, A. Itai, and A. Shamir, "On the complexity of timetable and multi-commodity flow problems," *Proc. - Annu. IEEE Symp. Found. Comput. Sci. FOCS*, vol. 1975-Octob, pp. 184–193, 1975.

[6] M. J. F. Souza, N. Maculan, and L. S. Ochi, "A GRASP-Tabu Search Algorithm For Solving School Timetabling Problems," *Metaheuristics Comput. Decis.*, no. June 2014, p. 707, 2004.

[7] G. H. G. Fonseca, H. G. Santos, T. Â. M. Toffolo, S. S. Brito, and M. J. F. Souza, "GOAL solver: a hybrid local search based solver for high school timetabling," *Ann. Oper. Res.*, vol. 239, no. 1, pp. 77–97, 2014.

[8] L. N. Ahmed, E. Özcan, and A. Kheiri, "Solving high school timetabling problems worldwide using selection hyper-heuristics," *Expert Syst. Appl.*, vol. 42, no. 13, pp. 5463–5471, 2015.

[9] L. Saviniec, M. O. Santos, and A. M. Costa, "Parallel local search algorithms for high school timetabling problems," *Eur. J. Oper. Res.*, vol. 265, no. 1, pp. 81–98, 2017.

[10] S. Even, A. Itai, and A. Shamir, "On the Complexity of Timetable and Multicommodity Flow Problems," *SIAM J. Comput.*, vol. 5, no. 4, pp. 691–703, 1976.

[11] J. H. Kingston, "KHE14: An Algorithm for High School Timetabling," *Proc. Tenth Int. Conf. Pract. Theory Autom. Timetabling*, no. August 2014, pp. 269–291, 2014.

[12] J. H. Kingston, "A tiling algorithm for high school timetabling," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 3616 LNCS, pp. 208–225, 2005.

[13] J. H. Kingston, "Hierarchical Timetable Construction," in *Practice and Theory of Au- tomated Timetabling VI (Springer Lecture Notes in Computer Science 3867)*, 2007, pp. 294–307.

[14] J. H. Kingston, "Resource assignment in high school timetabling," *Ann. Oper. Res.*, vol. 194, no. 1, pp. 241–254, 2012.

[15] G. Post *et al.*, "An XML format for benchmarks in High School Timetabling," *Ann Oper Res*, vol. 194, pp. 385–397, 2012.

[16] G. Post *et al.*, "XHSTT: An XML archive for high school timetabling problems in different countries," *Ann. Oper. Res.*, vol. 218, no. 1, pp. 295–301, 2014.

[17] A. Kheiri, E. Özcan, and A. J. Parkes, "A stochastic local search algorithm with adaptive acceptance for high-school timetabling," *Ann. Oper. Res.*, vol. 239, no. 1, pp. 135–151, 2016.

[18] I. X. Tassopoulos, C. A. Iliopoulou, and G. N. Beligiannis, "Solving the Greek school timetabling problem by a mixed integer programming model," *J. Oper. Res. Soc.*, vol. 0, no. 0, pp. 1–16, 2019.

[19] J. Kingston, "A Software Library for High School Timetabling and Nurse Rostering," 2020. [Online]. Available: http://jeffreykingston.id.au/khe/. [Accessed: 26-Jan-2020].

[20] G. H. G. Fonseca, H. G. Santos, T. Â. M. Toffolo, S. S. Brito, and M. J. F. Souza, "GOAL solver: a hybrid local search based solver for high school timetabling," *Ann. Oper. Res.*, vol. 239, no. 1, pp. 77–97, 2016.

[21] Brilliant.org, "Markov Chains," 2020. [Online]. Available: https://brilliant.org/wiki/markov-chains/. [Accessed: 26-Jan-2020].

[22] X.-R. Cao, "Perturbation Analysis," in *Stochastic Learning and Optimization: A Sensitivity-Based Approach*, Boston, MA: Springer US, 2007, pp. 51–146.

[23] G. H. G. Fonseca, H. G. Santos, and E. G. Carrano, "Integrating matheuristics and metaheuristics for timetabling," *Comput. Oper. Res.*, vol. 74, pp. 108–117, 2016.