

5th International Conference on Soft Computing, Intelligent System and Information Technology



INFORMATICS DEPARTMENT Petra Christian University

CERTIFICATE

This certificate is presented to

ADE JAMAL

in recognition as

PRESENTER

in the

Intelligent System and Information Technology (ICSIIT 2017) 5th International Conference on Soft Computing,

26-29 September 2017, Bali, Indonesia



International Conference on Soft Computing Intelligent System and Information Technology









T. Catalyst Solusi Integrasi

okopedia

Supported by

General Chair

Proceedings

2017 International Conference on Soft Computing, Intelligent System and Information Technology ICSIIT 2017

Denpasar, Bali, Indonesia 26-29 September 2017

Edited by Henry Novianus Palit and Leo Willyanto Santoso



IEEE Computer Society Technical & Conference Activities Board

T&C Board Vice President

Hausi Müller University of Victoria, Canada

IEEE Computer Society Staff

Evan Butterfield, Director of Products and Services Patrick Kellenberger, Manager, Conference Publishing Services

IEEE Computer Society Publications

The world-renowned IEEE Computer Society publishes, promotes, and distributes a wide variety of authoritative computer science and engineering texts. These books are available from most retail outlets. Visit the CS Store at *http://www.computer.org/portal/site/store/index.jsp* for a list of products.

IEEE Computer Society Conference Publishing Services (CPS)

The IEEE Computer Society produces conference publications for more than 300 acclaimed international conferences each year in a variety of formats, including books, CD-ROMs, USB Drives, and on-line publications. For information about the IEEE Computer Society's *Conference Publishing Services* (CPS), please e-mail: cps@computer.org or telephone +1-714-821-8380. Fax +1-714-761-1784. Additional information about *Conference Publishing Services* (CPS) can be accessed from our web site at: *http://www.computer.org/cps*

Revised: 18 January 2012



CPS Online is our innovative online collaborative conference publishing system designed to speed the delivery of price quotations and provide conferences with real-time access to all of a project's publication materials during production, including the final papers. The **CPS Online** workspace gives a conference the opportunity to upload files through any Web browser, check status and scheduling on their project, make changes to the Table of Contents and Front Matter, approve editorial changes and proofs, and communicate with their CPS editor through discussion forums, chat tools, commenting tools and e-mail.

The following is the URL link to the *CPS Online* Publishing Inquiry Form: http://www.computer.org/portal/web/cscps/quote

2017 International Conference on Soft Computing, Intelligent System and Information Technology

ICSIIT 2017

Table of Contents

Preface	xi
Conference Organization	xii
Program Committee	xiii
Reviewers	xv

Classification and Correlation Techniques

Gesture Recognition for Indonesian Sign Language Systems (ISLS) Using Multimodal Sensor Leap Motion and Myo Armband Controllers Based on Naïve Bayes Classifier
Khamid, Adhi Dharma Wibawa, and Surya Sumpeno
Waah: Infants Cry Classification of Physiological State Based on Audio Features
Ramon L. Rodriguez and Susan S. Caluya
Fuzzy Clustering and Bidirectional Long Short-Term Memory for Sleep Stages Classification
Intan Nurma Yulita, Mohamad Ivan Fanany, and Aniati Murni Arymurthy
MFCC Feature Classification from Culex and Aedes Aegypti Mosquitoes Noise Using Support Vector Machine
Achmad Lukman, Agus Harjoko, and Chuan-Kay Yang
Automatic Chord Arrangement with Key Detection for Monophonic Music
Credit Scoring Refinement Using Optimized Logistic Regression
Anomaly Detection System Based on Classifier Fusion in ICS Environment

Efficient Object Recognition with Multi-directional Features in Urban Scenes	
Ryo Kawanami and Kousuke Matsushima	

Feature Extraction and Image Recognition Methods

The Model and Implementation of Javanese Script Image Transliteration
Anastasia Rita Widiarti, Agus Harjoko, Marsono, and Sri Hartati
Human Activity Recognition by Using Nearest Neighbor Algorithm from Digital Image
Muhammad Ihsan Zul, Istianah Muslim, and Luqman Hakim Night to Day Algorithm for Video Camera
Arca Detection and Matching Using Scale Invariant Feature Transform (SIFT) Method of Stereo Camera Aviv Yuniar Rahman, Surya Sumpeno, and Mauridhi Hery Purnomo
The Application of Deep Convolutional Denoising Autoencoder for Optical Character Recognition Preprocessing
Acne Segmentation and Classification Using Region Growing and Self-Organizing Map

Algorithms for Intelligent Computation

Extended Concept of Generalized Fuzzy Rough Sets on Asymmetric Fuzzy	
Coverings	84
The Proposal of the Software for the Soft Targets Assessment	00
Lucia Duricova, Martin Hromada, and Jan Mrazek	
Application of Artificial Intelligence (AI) in Search Engine Optimization (SEO) Yodhi Yuniarthe	96
Spatial Model Design of Landslide Vulnerability Early Detection	
with Exponential Smoothing Method Using Google API Kristoko Dwi Hartomo, Sri Yulianto, and Joko Ma'ruf	102
Measures of Dependency in Metric Decision Systems and Databases Anh Duy Tran, Somjit Arch-int, and Ngamnij Arch-int	107
Multiple Scattered Local Search for Course Scheduling Problem	114

The Software Proposes for Management and Decision Making at Process Transportation Jan Mrazek, Lucia Duricova, and Martin Hromada	.120
Distributed Systems and Computer Networks	
A Self-Adaptive Architecture with Energy Management in Virtualized Environments	.124
l Made Murwantara, Behzad Bordbar, and João Bosco Ferreira Filho	
Nanoservices as Generalization Services in Service-Oriented Architecture	.131
Automated Concurrency Testing for Cloud-Based Polling Systems	.138
Low-Overhead Multihop Device-to-Device Communications in Software Defined Wireless Networks <i>Riyanto Jayadi and Yuan-Cheng Lai</i>	.144
A Secure Anonymous Authentication Scheme for Roaming Service in Global Mobility Network	.150
Linux PAM to LDAP Authentication Migration Justinus Andjarwirawan, Henry Novianus Palit, and Julio Christian Salim	.155
Exploratory Research on Developing Hadoop-Based Data Analytics Tools Henry Novianus Palit, Lily Puspa Dewi, Andreas Handojo, Kenny Basuki, and Mikiavonty Endrawati Mirabel	.160

Mobile and Pervasive IoT Applications

Human Heart Rate Detection Application Semuil Tjiharjadi and Aufar Fajar	167
Near Field Communication Technology in Delivering Information in Museums Djoni Haryadi Setiabudi, Ryan Christian Wiguno, and Henry Novianus Palit	173
Android Application for Monitoring Soil Moisture Using Raspberry Pi Lily Puspa Dewi, Justinus Andjarwirawan, and Robin Putra Wardojo	178
Development of Mobile Indoor Positioning System Application Using Android and Bluetooth Low Energy with Trilateration Method <i>Agustinus Noertjahyana, Ignatius Alex Wijayanto, and Justinus Andjarwirawan</i>	185

Assessments of Integrated IS/IT

The Proposal of United Crisis Management Information Systems of the Czech Republic1	90
Katerina Vichova, Martin Hromada, and Ludek Lukas	
The Analysis of Academic Information System Success: A Case Study at Instituto Profissional De Canossa (IPDC) Dili Timor-Leste1 Apolinario Dos Santos, Albertus Joko Santoso, and Djoko Budiyanto Setyohadi	96
Identification of Factors Influencing the Success of Hospital Information System (SIRS) by Hot-Fit Model 2006: A Case Study of RSUD Dr Samratulangi Tondano, Minahasa Regency, North Sulawesi2	202
Frendy Rocky Rumambi, Albertus Joko Santoso, and Djoko Budyanto Setyohadi	
The Alignment of IT and Business Strategy at ROC Leeuwenborgh2 Frederick Wonges, Jack Zijlmans, and Leo Willyanto Santoso	208
Development of Capability Assessment Model of IT Operation Management Process with Organizational Behavior2 Luh Made Wisnu Satyaninggrat and Kridanto Surendro	214
Exploring Critical Success Factors of Mobile Learning as Perceived by Students of the College of Computer Studies – National University	20
Identifying Characteristics and Configurations in Open Source ERP in Accounting Using ASAP: A Case Study on SME2 Agung Terminanto and Achmad Nizar Hidayanto	27

Simulation and Virtual Reality Applications

The Real Time Training System with Kinect: Trainer Approach Ivana Valentine Masala and Apriandy Angdresey	.233
3D LIDAR City Model Application and Marketing Plan Development	.238
Periodic Review Inventory Model Simulations for Imperfect Quality Items and Stochastic Demand Gede A. Widyadana, Audrey T. Widjaja, and Irena Liong	.243
Simulation on Crowd Mobility of Moving Objects Using Multi-agent and ClearPath Baihaqi Siregar, Agnes Irene Silitonga, Erna Budhiarti Nababan, Ulfi Andayani, and Fahmi Fahmi	.250
Truck Management Integrated Information System in a Shipping Line Company Arnold Samuel Chan and I Nyoman Sutapa	.257

Simulation of Atmosphere in Trowulan during the Golden Era of Majapahit	263
Daniel Kusuma, Rudi Adipranata, and Erandaru	203
Development of Interactive Learning Media for Simulating Human Digestive System	270
Kristo Radion Purba, Liliana, and Daniel Runtulalu	
Development of Interactive Learning Media for Simulating Human Blood Circulatory System	275
Kristo Radion Purba, Liliana, and Yohanes Nicolas Paulo Kwarrie	

Smart Assistive Technologies

Fall Detection Application Using Kinect	279
Driver Drowsiness Detection Using Visual Information on Android Device Aldila Riztiane, David Habsara Hareva, Dina Stefani, and Samuel Lukas	283
Epileptic Alert System on Smartphone Aziis Yudha Adwitiya, David Habsara Hareva, and Irene Astuti Lazarusli	288
Elderly Healthcare Assistance Application Using Mobile Phone Andreas Handojo, Tioe Julio Adrian Sutiono, and Anita Nathania Purbowo	292
Socially-Enhanced Variants of Mobile Bingo Game: Towards Personalized Cognitive and Social Engagement among Seniors	297

Smart Mobile Applications

M-Guide: Hybrid Recommender System Tourism in East-Timor Jaime da Costa Lobo Soares, Suyoto, and Albertus Joko Santoso	303
M-Guide: Recommending Systems of Food Centre in Buleleng Regency	310
Empowering Public Secondary Schools on Disaster Response and Recovery: A Framework for the Development of Helpline Mobile Application Odette Saavedra, Matthew C. Abrera, Mickaela Carla L. Waniwan, Curly Kale C. Dava, and Bernie S. Fabito	315
A Framework Mobile Game Application that Teaches Parts of Speech in Grade 3 in Filipino John Erasmus Correa, Jastine Gamboa, Mark Edison Lavapie, Edzel Uy, and Ramon L. Rodriguez	
iSagip: A Crowdsource Disaster Relief and Monitoring Application Framework Auxesis Jacobi M. Schwab, John Eduard C. Omaña, Kent V. Roazol,	

Ted Anthony Y. Abe, and Bernie S. Fabito

Case Studies of Knowledge Discovery and Management

Executive Dashboard as a Tool for Knowledge Discovery Nyoman Karna	331
Data Mining Applications for Sales Information System Using Market Basket Analysis on Stationery Company Alexander Setiawan, Gregorius Satia Budhi, Djoni Haryadi Setiabudi, and Ricky Djunaidy	337
A Knowledge Management-Extended Gamified Customer Relationship Management System <i>Chien-Sing Lee, Jun-Jie Foo, Vinudha a/p Jeya Sangar, Pei-Yee Chan,</i> <i>Weng-Keen Hor, and Eng-Keong Chan</i>	341
Web Based Customer Relationship Management Application for Helping Sales Analysis on Bike Manufacturer Anita Nathania Purbowo, Yulia, and Agustinus Ivan Suryadi	347
Replenishment Strategy Based on Historical Data and Forecast of Safety Stock	353
On Estimation and Prediction of Simple Model and Spatial Hierarchical Model for Temperature Extremes Indriati Njoto Bisono	359
Author Index	

Conference Organization ICSIIT 2017

General Chairs

Leo W. Santoso, *Petra Christian University, Indonesia* Rajkumar Buyya, *The University of Melbourne, Australia*

Program Chairs

Henry N. Palit, Petra Christian University, Indonesia Chia-Hui Chang, National Central University, Taiwan

Steering Committee

Rolly Intan, *Petra Christian University, Indonesia* Chi-Hung Chi, *CSIRO, Australia* Rajkumar Buyya, *The University of Melbourne, Australia*

Program Committee ICSIIT 2017

A.V. Senthil Kumar, Hindusthan College of Arts and Science, India Achmad Nizar Hidayanto, University of Indonesia, Indonesia Alexander Fridman, Institute for Informatics and Mathematical Modelling, Russia Ashraf Elnagar, University of Sharjah, United Arab Emirates Bernardo Nugroho Yahya, Hankuk University of Foreign Studies, Korea Bor-Shen Lin, National Taiwan University of Science and Technology, Taiwan Budi Bambang, Indonesia Bruce Spencer, University of New Brunswick, Canada Can Wang, CSIRO, Australia Chen Ding, Rverson University, Canada Cherry G. Ballangan, Australia Chuan-Kai Yang, National Taiwan University of Science and Technology, Taiwan Edwin Lughofer, Johannes Kepler University Linz, Austria Eric Holloway, Baylor University, USA Erma Suryani, Sepuluh Nopember Institute of Technology, Indonesia Felix Pasila, Petra Christian University, Indonesia Han-You Jeong, Pusan National University, Korea Hans Dulimarta, Grand Valley State University, USA Hong Xie, Murdoch University, Australia Ilung Pranata, SAP, Australia Jantima Polpinij, Mahasarakham University, Thailand Kassim S. Mwitondi, Sheffield Hallam University, United Kingdom K.V. Krishna Kishore, Vignan University, India Lee Chien Sing, Sunway University, Malaysia Mahmoud Farfoura, Royal Scientific Society, Jordan Masashi Emoto, Meiji University, Japan Moeljono Widjaja, Agency for the Assessment and Application of Technology, Indonesia Nai-Wei Lo, National Taiwan University of Science and Technology, Taiwan Prantosh Kumar Paul, Raiganj University, India Pujianto Yugopuspito, Pelita Harapan University, Indonesia Raymond Kosala, Binus University, Indonesia Rudy Setiono, National University of Singapore, Singapore Sankar Kumar Pal, Indian Statistical Institute, India Selpi, Chalmers University of Technology, Sweden Sergio Camorlinga, University of Winnipeg, Canada Shafiq Alam Burki, University of Auckland, New Zealand Shan-Ling Pan, University of New South Wales, Australia Simon Fong, University of Macau, Macau Smarajit Bose, Indian Statistical Institute, India Son Kuswadi, Electronic Engineering Polytechnic Institute of Surabaya, Indonesia Suphamit Chittayasothorn, King Mongkut's Institute of Technology Ladkrabang, Thailand Todorka Alexandrova, Waseda University, Japan Tohari Ahmad, Sepuluh Nopember Institute of Technology, Indonesia Vijay Varadharajan, Macquarie University, Australia Wei Zhou, CSIRO, Australia Wichian Chutimaskul, King Mongkut's University of Technology Thonburi, Thailand

Xiaojun Ye, Tsinghua University, China Xiao Wu, Southeast University, China Yanqing Liu, Jiangxi University of Finance and Economics, China Yung-Chen Hung, Soochow University, Taiwan Yunwei Zhao, Tsinghua University, China

Multiple Scattered Local Search for Course Scheduling Problem

Ade Jamal Department of Informatics Al-Azhar University of Indonesia Jakarta, Indonesia adja@uai.ac.id

Abstract—Course scheduling problem is a nondeterministic polynomial time-complete problem which is hard to solve. One of the popular method for solving the problem is the so-called local search algorithm family, which is basically search a better solution in the neighborhood of previously known potential solution iteratively. Local search technique has a weakness that it gets stuck easily in a local optimum or on a plateau. Various efforts have been done in many literatures to circumvent this weakness. Current work investigates an already improved scattered local search by augmenting randomly generated initial population of schedules in the local search algorithm. This algorithm will be compared to an originally population based genetic algorithm which is enhanced by letting local search technique on each individual schedule in every generation. The result has shown that the population based local search performs better than the hybrid genetic algorithm.

Keywords—course scheduling; optimization; local search algorithms; genetic algorithms

I. INTRODUCTION

Constructing a course schedule is one of the main challenge within a university that must be performed in every academic period. This problem is considered as a nondeterministic polynomial time-complete problem [1]–[2] which is quite difficult and time-consuming to solve. This is a multi-dimensional and multi-constrained combinatorial optimization problem, that has been being a subject of extensive research efforts due to its complexity and wide application such as school timetables [3], exam scheduling [2] and course scheduling [1], [4]–[8].

Course scheduling process involves placements of many events in available time and space resources where initially specified constraints must be satisfied. In this study, an event could be a teaching class or a laboratory class where a combination of lecturers or instructors, student-groups and a course was designated beforehand. This type of course scheduling is called a curriculum based course scheduling or timetabling wherein a student-group is associated to the curriculums [9]. Scheduling problem could also include selections of courses that should be taught by which lecturer. Another type of scheduling involves every individual student who has enrolled for their courses. These last two types of course scheduling are not considered in the present study. There are two sets of predefined constraints which must be deliberated by the university course scheduling problem. The first set of constraints is named hard constraints which must not be violated to construct a valid or feasible schedule. The second set is soft constraints which are desired but not necessarily to be fulfilled. Hence, finding the valid or feasible schedule is merely a search problem rather than an optimization problem. However, fulfilling the soft constraints is a real optimization process in this study.

Course scheduling problem has been solved by wide range type of algorithms. The most traditional one is a socalled graph coloring heuristic method whereby courses are assigned to rooms and time-slots one by one in particular order. The second type falls into the local search algorithm family [8] whereby searching is carried out in neighborhood of a known state rather than exploring search space extensively. This type of algorithm is quite popular among the researcher who recommended enhanced variant of local search algorithm such as simulated annealing method [3], [5], tabu search method [4] and hill-climbing search method [6].

The last type of algorithm is based on population mechanism which starts with many different solutions and explores possible solution in wider search space. The most popular population based algorithms used to solve course scheduling are evolutionary algorithms [10], genetic algorithm [2], [7], [9], and harmony algorithm [1].

Due to two different criteria in the course scheduling problem, i.e. hard constraints that govern the search problem and soft constraints that constitute the objective function of the optimization problem, various approaches can be found in the published research works. Combining these two set constraints with weight factors is the most frequently used approach, wherein the weight factor of soft constraint is just small fraction of hard constraint weight factor. Another popular approach is a staging method wherein a fulfilling of hard constraints in the first stage followed by an optimization process to gain as many as possible soft constraint satisfactions [7]. Some of the researchers used hybrid algorithms to tackle this staging method [9], [11].

In the previous study, we had incorporated evolutionary algorithm (EA) [12] and the improved scattered hill-climbing search method (SHC) [6] into three stage method [11]. We had also compared these two algorithms separately, and proved that SHC is faster than EA in case SHC could find the feasible schedule [13]. Though, the probability of success in searching the feasible solution, i.e. the hard constraint free schedule is not more than 30% for SHC. This is caused by the nature of local search of SHC which is good in exploiting the possible solution in the neighborhood of a solution state. In contrary, the EA is good in exploring possible solution by generating new successor state by mutation and cross-over two parent states. Hence the probability of success in searching the feasible schedule is much better, but it needs much more iteration, i.e. number of generation because the nature of probabilistic process in EA.

Knowing the above mentioned phenomenon, a number of SHC methods from randomly generated initial states will be conducted to obtain higher probability of success in finding feasible schedules. In case of EA, the exploitation capability will be embedded in each member of population before the evolution process started in every new generation.

II. SOLUTION METHOD FOR COURSE SCHEDULING PROBLEM

A. Course Scheduling Problem

The curriculum based course scheduling problem consist of the following entities:

- A set of room {*r*₁, *r*₂, ... *r*_{noR}} which has a seat capacity and specified as a lecture- or an interactive computer classroom,
- A set of time-slot {*t*₁, *t*₂, ... *t*_{noT}} wherein lecturers has preferences to teach,
- A set of course {*c*₁, *c*₂, ... *c*_{noC}} which is based on curriculum for each student group and may have multiple section (i.e. credit unit where one course section occupies more than one time-slot),
- A set of lecturer { l_1 , l_2 , ..., l_{noL} } which has been assigned to specific course(s) and has a certain unavailable time-slot and preference time-slot,
- A set of student group $\{s_1, s_2, \dots, s_{noS}\}$ with associated number of student from a specific program and same grade,
- A set of events (i.e. classes) {*e*₁, *e*₂, ... *e*_{noE}}, to be scheduled in a certain number of time-slots and a room, is defined as combination of a specific course with assigned lecturer attended by certain number of student group.

A course schedule is said to be feasible and valid when the following hard constraints are satisfied:

- All events have been assigned to a time-slot and a room (complete schedule),
- No room is occupied by more than one event at the same time (room conflict),
- No lecturer teaches more than one event at the same time (lecturer conflict),
- No student group from the same program and same grade attends more than one event at the same time (student conflict),
- Number of attending students in the student group must not exceed the room capacity,

- Room has a feature (e.g. Laboratory) required by the assigned course,
- No lecturer teaches in time-slot which is unavailable for him/her,
- An event with a multiple section course must be assigned in the same room contiguously (continue event).

There are soft constraints that are preferably fulfilled to get a so-called optimum schedule. Some examples of these soft constraints are as follows:

- A lecturer should be assigned in his/her preference time-slot
- A lecturer should not be assigned in his/her avoiding time-slot
- A classroom should not be half empty, i.e. the number of attending students is less than a half of room capacity
- No student should be scheduled to sit more than three events on the same day
- Minimize scheduled events in the last time-slot of a day
- Morning class-hours are preferred than afternoon class-hours
- Minimize unoccupied room between two occupied time-slots

The first three of the above mentioned soft constraints will be taken into account in the presented work.

B. Course Scheduling Model

Course schedules are represented in a variety of models. References [1], [2], [7], [10] uses a two dimensional matrix where rooms and timeslots are represented by row and column and each cell of matrix represents a single event. In the previously published works [6], [11]–[13] we extended the dimensional matrix into three dimensional work by remodeling the one dimensional timeslot representing column into a two dimensional matrix where each row corresponds to day and column to hour. Another approach is direct representation, i.e. each gene is represented by a triple of event, room and time-slot $<e_i, r_j, t_k > [8], [10]$.

The matrix model governs that the "room conflict" hard constraint is always fulfilled. However, generating the initial state is quite difficult when all events must be assigned into the cells taken into account multiple section event must be assigned in the same room contiguously. Therefore, the matrix model is not applicable for the present work since it requires randomly generated initial state for SHC. However, instead of using direct representation of triple model, we introduce a tuple of two:

$$\langle e_{i}, f(r_{j}, t_{k}) \rangle \tag{1}$$

where space (room) and time resources are formulated as a single value function of resources $f(r_j, t_k)$ whose value ranging from 1 to multiplication of *noR* and *noT*, i.e. number of rooms and timeslots, respectively. Generating random initial state of schedule is as easy as generating random number *f* in this range for each event e_i .

C. Scheduling Algorithm

The original hill-climbing search method is a not population based technique. The fore knowledge that this kind of local search method gets stuck frequently in a local optimum as we had shown in [13], lead us to invoke a population of randomly generated schedules performing the scattered hill-climbing search method. This population based scattered hill climbing search, shortly hereafter named a multiple scattered local search, will proceed for certain number of iterations until one or more individual schedules from the population become feasible and valid, i.e. none of hard constraints are violated. While the iteration is still in progress, course schedules that already feasible will further undergo in the second stage, searching for the optimum solution in the neighborhood of the found feasible state.

Algorithm for a one-step single scattered hill-climbing search is shown in Fig. 1.

```
function SCATTERED-HILL-CLIMBING(state)
  inputs :
                    state : a schedule
  local variables : new, neighbor : a schedule
                    penalty : hard constraint violation
  returns :
                    new state which has less violation
                    on hard constraint
  penalty \leftarrow HC-SC-PENALTY(state)
  new \leftarrow state
  for i \leftarrow 1 to neighborhood size do
      neighbor \leftarrow MUTATE(state)
      if HC-SC-PENALTY(neighbor) ≤ penalty do
          new \leftarrow neighbor
          penalty \leftarrow HC-SC-PENALTY(neighbor)
      end if
  end for
  return new
```

Figure 1. Scattered hill-climbing algorithm

In the present work, this function is invoked for each schedule in the population. Note that the function of HC-SC-penalty is a combination of hard constraint violation (*penalty_{hc}*) reduced by a small part of soft constraint fulfillment (*score_{sc}*) as given in (2). The soft constraint fulfillment is an optional in this part of algorithm but it turns out that its existence is very significant to improve the effectiveness of the algorithm.

HC-SC-penalty = weight_{hc} * penalty_{hc} - weight_{sc} * score_{sc} (2)

Any schedule that reached the zero violation of hard constraint, will further undergo a local search algorithm as depicted in Fig. 2 to maximize the fulfillment of soft constraints (SC-SCORE). The multiple scattered local search iteration will stop when a specified number of schedules become feasible or maximum iteration is reached.

Performing the scattered local search in a population scheme rather than a single schedule gives more explorative capability in this local search method, of course at cost of extra computation time. To gain a better view whether this extra cost worthy, we will compare the performance of MSLS (multiple scattered local search) with the evolutionary algorithm [12] using the same two tuple model. We add also some exploitation booster by letting each chromosome at every new generation in the evolution process does the same single scattered hill-climbing search from Fig. 1. This addition of hill-climbing search into the standard genetic algorithm is also used in [14] to minimize two objective functions, i.e. soft constraint and robustness.

```
function LOCAL-SEARCH-OPTIMIZATION(state)
  inputs :
                    state : a feasible schedule
  local variables : new, neighbor : a schedule
                    score : soft constraint fulfillment
  returns :
                    new state which has more
                    fulfillment on soft constraint
  score \leftarrow SC-SCORE(state)
  new \leftarrow state
  for i \leftarrow 1 to neighborhood size do
      neighbor \leftarrow MUTATE(state)
      if SC-SCORE(neighbor) > score and
          HC-PENALTY(neighbor) = 0 do
          new \leftarrow neighbor
          score \leftarrow SC-SCORE(neighbor)
      end if
  end for
  return new
```

Figure 2. Scattered local search around the feasible schedule algortihm

Comparing with the previous work in [12], the algorithm in the present work can do the real genetic mechanism, namely cross-over. Because of the two tuple model from Eq. 1, the chromosome will form just like a common integer chromosome, as shown in Fig. 3. All events are always taken in the process, satisfy the complete schedule of hard constraint, hence this will not be destroyed by cross-over mechanism as it was in the three dimensional matrix model we used in [12]–[13]. The present evolutionary algorithm will be called hybrid genetic algorithm (HGA).



Figure 3. Course schedule chromosome

where

$$f_i \in \mathfrak{I}^{\text{number of space-time resources}}$$
 (3)

Tournament of 5 is used for selection of parent chromosome for crossover mechanism. Before parent selection, elitism mechanism is done by using truncated selection scheme. The selection is performed based on violation of hard constraint and fulfillment of soft constraint, combined using weight factor as given in (2).

Elite chromosomes that reach zero hard constraint violation, i.e. feasible schedule, will be further proceeded in the second phase performing local search optimization related to the soft constraints by keeping the zero violation of hard constraint as given in Fig. 2.

III. RESULTS AND DISCUSSION

Computation complexity of one iteration for these two different algorithms, i.e. Multiple Scattered Local Search (MSLS) and Hybrid Genetic Algorithm (HGA) will be discussed first. The complexity of evolutionary algorithm is more than two times of a single scattered hill-climbing algorithm when the population size of evolutionary algorithm and neighborhood size of hill-climbing are the same [13]. In the present work, the complexity of one iteration for MSLS and HGA is in the same order if the same population and neighborhood size are used in both algorithms. Hence, the number of iteration is the only extent of computation complexity when we compare MSLS and HGA with the same size of population and neighbors.

The probabilistic nature of both algorithms makes the behavior of algorithm difficult to analyze. References [15]–[16] introduced an empirical approach to analyze the behavior of non-deterministic algorithm by constructing Run Time Distribution (RTD) and Run Length Distribution (RLD). In practice, this empirical RTD and RLD are determined by running the respective algorithm for a number times on a given problem instance up to some cut-off time or iteration and then for each successful run, recording the required time or the required number of iteration to find a solution, respectively for RTD or RLD. For this study, we use two small sets of curriculum for experiment as given in Table I.

TABLE I. TWO SMALL SETS OF CURRICULUM DATA

Specification	Ι	II
Number of events	25	51
Number of instructors	14	23
Number of class rooms	2	4
Number of student groups	4	7
Number of events hour	67	138
Available time slots	80	160

For the first test, we run both algorithms to find a feasible schedule on the same curriculum data I by setting a cut-off iteration of 1000. The computational parameter size, i.e. population- and neighborhood size for both algorithms are set to be equal. The resulting RTD and RLD are given in Fig. 4 and Fig. 5. That RTD has similar shape as RLD proves the time consumed for one iteration by both algorithms with the same computational size parameters are equal. Furthermore, the test results have shown that MSLS yields higher probabilities of success for all range of iterations. This outcome is very contradictive with the previous comparing study between traditional GA and the single SHC wherein the first algorithm is more superior in the probability behavior [12]–[13].

Both computational size parameters give the extent of randomization on the algorithms but for different objectives. While the number of schedule population aims to add more exploration power in a wider search area, the neighborhood size of scattered schedules yields more exploitation random capability in depth.

Effect of these two computational parameter sizes are studied. Higher neighborhood size yields better probability in term of iterations as shown in RLD from Fig. 6. However, in term of run time, this parameter has no effect since this neighborhood size is inversely proportional to run time as depicted in RTD from Fig.7.







Figure 5. RLD for MSLS and HGA with the same computational size parameter, i.e. population size = 40 and neighborhood size = 50



Figure 6. RLD for MSLS with the same population size = 40 and variation of neighborhood size (n = 5, 25, 50)

Run time distribution



Figure 7. RTD for MSLS with the same population size = 40 and variation of neighborhood size (n = 5, 25, 50)

Effect of the population sizes are presented in Fig. 8 and Fig. 9. Note that when only one schedule is considered in the population, namely Sc=1, MSLS becomes a scattered local search as in the previous published works [6], [13]. The probability of success for the scattered local search here is not more than 50%. Probability in the iteration distribution is getting better when the population size is larger, though in run time distribution probability curve shifts right to slower region of computing run time. In spite of this, it is proven that a probability of 99% success for MSLS needs a sufficient population size.

Results given in Fig. 8 and Fig. 9 are obtained by running MSLS on curriculum data II for 200 times up to cut-off 5000 iterations. The problem of curriculum data II is two times bigger than data I. Doubling the problem size requires at least five times number of iterations and eight times longer run time for a same level of probability of success.

Beside the size parameters, i.e. population- and neighborhood size, MSLS has also two weight factors to be set. In searching for feasible schedules, penalty function that includes small contribution of soft constraint satisfaction, i.e. 0.1 SC + 0.9 HC yields better probability curve than 100% HC in the penalty function as shown in Fig. 10. Nevertheless, excessive contribution of soft constraint satisfaction will shift RTD curve slightly to slower region of computation time.



Figure 8. RLD for MSLS with the same neighborhood size = 50 and variation of population size ($S_c = 1, 5, 40$)



Figure 9. RTD for MSLS with the same neighborhood size = 50 and variation of population size ($S_c = 1, 5, 40$)



Figure 10. RTD for MSLS with the same neighborhood size = 50 and population size = 40, varying weight factor in penalty function in (2)

RTD shown in Fig. 10 is obtained from 200 test runs on curriculum data I up to cut-off 100 iterations or four feasible schedules were found. It is interesting to note here that MSLS is capable to seek a number of valid schedules up to about 20% of population size without too much extra iterations or computation time. RLD from this test is not presented because it has similar shape with RTD since run time required for one iteration is equal in this particular case for every probability curve in Fig. 10.



Figure 11. RLD for HGA with the same population size = 40 and variation of neighborhood size (n = 0, 2, 4, 50)



Figure 12. RTD for HGA with the same population size = 40 and variation of neighborhood size (n = 0, 2, 4, 50)

For the last test, we evaluate effect of neighborhood size on the probabilistic of success for HGA. The test results are given again in Fig. 11 and Fig. 12. If neighborhood size n=0, HGA becomes a traditional GA, and the test result shows that this GA is the worst among HGA with respect to RTD and RLD, even for HGA with very small neighborhood n=2. Similar to the case for MSLS, effect of neighborhood size only significant on RLD, not on RTD with the same reason.

IV. CONCLUSIONS

We have presented a multiple scattered local search algorithm, which is a population based version of randomized hill-climbing algorithm. Compared to the original hill-climbing search, the algorithm incorporates stochastic behavior in two different parts, namely during the improvement step in the neighborhood region and in the constructing of initial population states. The probabilistic behavior of the algorithm was evaluated using empirical approaches by constructing run time distribution and run length distribution. The test results have shown that the probabilistic of success run for the presented algorithm is better than the original scattered local search and also much better than the hybrid genetic algorithm for the same problem instances.

Another advantage of the multiple scattered local search which is still to be exploited, is a strong parallelism inherent in the algorithm. Taking advantage of this parallelism by implementing the algorithm in parallel processing is practically necessary to solve the real world course scheduling problem whose problem size much bigger than in the presented work.

ACKNOWLEDGMENT

This article is sponsored by LP2M UAI through International Seminar Grant.

REFERENCES

- M.A. Al-Betar, A.T. Khader, and T.A. Gani, "A Harmony Search Algorithm for University Course Timetabling," Proc. 7th Int. Conf. on the Practice and Theory of Automated Timetabling (PATAT), Montréal (Canada), Aug. 2008, Ann. Operations Research, vol. 194, no. 1, E. Burke, M. Gendreau, Eds., pp. 3–31, Apr. 2012, doi: 10.1007/s10479-010-0769-z.
- [2] E.K. Burke, D.G. Elliman and R. Weare, "A Genetic Algorithm Based University Timetabling System," Proc. 2nd East-West Int. Conf. on Computer Technologies in Education, Crimea (Ukraine), Sep. 1994, , pp. 35–40.
- [3] D. Abramson, "Constructing School Timetables Using Simulated Annealing: Sequential and Parallel Algorithms," Management

Science, vol. 37, no. 1, pp. 98-113, Jan. 1991, doi: 10.1287/mnsc.37.1.98.

- [4] A. Elloumi, H. Kamoun, J. Ferland, and A. Dammak, "A Tabu Search for Course Timetabling Problem at a Tunisian," Proc. 7th Int. Conf. on the Practice and Theory of Automated Timetabling (PATAT), Montréal (Canada), Aug. 2008.
- [5] M.A.S. Elmohamed, P. Coddington, and G. Fox, "A Comparison of Annealing Techniques for Academic Course Scheduling," Proc. 2nd Int. Conf. on the Practice and Theory of Automated Timetabling (PATAT), Toronto (Canada), Aug. 1997, LNCS, vol. 1408, E. Burke and M. Carter, Eds., pp. 92–112, doi: 10.1007/BFb0055883.
- [6] A. Jamal, "Solving University Course Scheduling Problem Using Improved Hill Climbing Approach," Proc. Int. Joint Seminar in Engineering (IJSE), Jakarta (Indonesia), Aug. 2008.
- [7] R. Lewis and B. Paechter, "Application of the Grouping Genetic Algorithm to University Course Timetabling," Proc. 5th European Conf. on Evolutionary Computation in Combinatorial Optimization (EvoCOP), Lausanne (Switzerland), Mar. 2005, LNCS, vol. 3448, G.R. Raidl and J. Gottlieb, Eds., pp. 144–153, doi: 10.1007/978-3-540-31996-2_14.
- [8] D. Moody, G. Kendall, and A. Bar-Noy, "Constructing Initial Neighborhoods to Identify Critical Constraints," Proc. 7th Int. Conf. on the Practice and Theory of Automated Timetabling (PATAT), Montréal (Canada), Aug. 2008.
- [9] S. Massoodian, A. Esteki, "A Hybrid Genetic Algorithm for Curriculum Based Course Timetabling Problem," Proc. 7th Int. Conf. on the Practice and Theory of Automated Timetabling (PATAT), Montréal (Canada), Aug. 2008.
- [10] P. Myszkowski and M. Norberciak, "Evolutionary Algorithms for Timetable Problems," Annales UMCS Sectio AI Informatica, vol. 1, no. 1, pp. 115–125, 2003.
- [11] A. Jamal, "A Three Stages Approach of Evolutionary Algorithm and Local-Search for Solving Hard and Soft Constrained Course Scheduling Problem," Proc. 11th Seminar on Intelligent Technology and Its Application (SITIA), Surabaya (Indonesia), Oct. 2010, paper no. 118, pp. 324–328.
- [12] A. Jamal, "University Course Scheduling Using the Evolutionary Algorithm," Proc. 2nd Int. Conf. on Soft Computing, Intelligent System and Information Technology (ICSIIT), Bali (Indonesia), Jul. 2010, pp. 86–90. (Republished in Jurnal Al-Azhar Indonesia, Seri Sains dan Teknologi, vol. 1, no. 1, pp. 7–12, Mar. 2011)
- [13] A. Jamal, "Evaluation of Modified Scattered Hill-Climbing Method and Evolutionary Approach for Course Scheduling Problem," Prosiding Seminar Nasional Komputasi (SNAKOM) [Proc. National Seminar in Computation], Bandung (Indonesia), Sep. 2012.
- [14] C. Akkan and A. Gülcü, "A Bi-Criteria Hybrid Genetic Algorithm with Robustness Objective for the Course Timetabling Problem," Proc. 11th Int. Conf. on Practice and Theory of Automated Timetabling (PATAT), Udine (Italy), Aug. 2016, pp. 451–456.
- [15] H.H. Hoos and T. Stützle, Stochastic Local Search: Foundations and Applications, Morgan Kaufmann, 2005
- [16] H.H. Hoos, "Stochastic Local Search Methods, Models, Applications", Ph.D. Dissertation, Department of Computer Science, Technischen Universität Darmstadt, Darmstadt (Germany), 1998.