Fairness in academic course timetabling

Moritz Mühlenhthalere · Rolf Wanka

Abstract We consider the problem of creating fair course timetables in the setting of a university. The central idea is that undesirable arrangements in the course timetable, i.e., violations of soft constraints, should be distributed in a fair way among the stakeholders. We propose and discuss in detail two fair versions of the popular curriculum-based course timetabling (CB-CTT) problem, the MMF-CB-CTT problem and the JFI-CB-CTT problem, which are based on max–min fairness (MMF) and Jain’s fairness index (JFI), respectively. For solving the MMF-CB-CTT problem, we present and experimentally evaluate an optimization algorithm based on simulated annealing. We introduce three different energy difference measures and evaluate their impact on the overall algorithm performance. The proposed algorithm improves the fairness on 20 out of 32 standard instances compared to the known best timetables. The JFI-CB-CTT problem formulation focuses on the trade-off between fairness and the aggregated soft constraint violations. Here, our experimental evaluation shows that the known best solutions to 32 CB-CTT standard instances are quite fair with respect to JFI. Our experiments show that the fairness can often be improved at the cost of only a small increase in the overall amount of penalty.

Keywords Curriculum-based course timetabling · Max–min fairness · Fairness index

1 Introduction

The university course timetabling problem (UCTP) captures the task of assigning courses to a limited number of resources (rooms and timeslots) in the setting of a university. In this

This is the extended version of Mühlenhthalere and Wanka (2012) presented on PATAT 2012.

M. Mühlenhthalere · R. Wanka
Department of Computer Science, University of Erlangen-Nuremberg, Cauerstrasse 11, 90459 Erlangen, Germany
e-mail: moritz.muehlenhthaler@cs.fau.de

R. Wanka
e-mail: rolf.wanka@cs.fau.de

© Springer Science+Business Media New York 2014
work, we consider the problem of creating fair course timetables for a particular variant of
the UCTP, the curriculum-based course timetabling (CB-CTT) problem proposed in (Di
Gaspero et al. 2007). The problem features various hard and soft constraints which model
typical real-world requirements at a university. Each violation of a soft constraint results in
a penalty for the timetable, and the task is to find a timetable such that the sum of the
penalties is minimal and all hard constraints are satisfied. However, this approach does not
consider the distribution of penalty over the different stakeholders, the students and
teachers. Situations may arise, in which a large part of penalty hits only a small group of
the stakeholders, who would thus receive a poor timetable in comparison to others. In
the context of examination timetabling, Bullnheimer has argued that minimizing the sum of
the penalties “might yield hard consequences for some students” (Bullnheimer 1998). In
order to achieve a fair distribution of penalty over the stakeholders, we include fairness
criteria in the objective function of the CB-CTT problem.

Fairness has been studied in various areas and there are certainly many valid ways to
characterize fair distributions depending on the particular application. For instance, we
may consider the total amount of allocated resources, the outcome for the worst-off
stakeholder, the deviation from the mean allocation, and so forth, see for example
(Ogryczak 2010; Soomer and Koole 2008; Feldman and Serrano 2006). We propose two
fair variants of the CB-CTT problem, which differ with respect to the underlying notion of
fairness. Our first problem formulation, the max–min fair curriculum-based course
timetabling (MMF-CB-CTT) problem, is based on lexicographic max-min fairness
(MMF). MMF aims at producing an optimal outcome for the worst-off stakeholder. Under
this condition, the outcome for the second worst-off stakeholder should be optimal as well,
and so on in this order for all remaining stakeholders. This fairness notion often appears in
the context of network bandwidth allocation (e.g., Bertsekas and Gallager 1992; Salles and
Barria 2008). MMF is a purely qualitative notion of fairness, i.e., given two allocations, we
can determine which allocation is fairer, but not by how much.

In order to solve the MMF-CB-CTT problem, we propose MAXMINFAIR_SA, an opti-
mization algorithm based on simulated annealing (SA). Due to the mild requirements of
SA on the problem structure, the proposed algorithm can easily be tailored to other max–
min fair optimization problems. A delicate part of the algorithm is the energy difference
function, which quantifies how much worse one solution is compared to another solution—
a piece of information that is not contained in the MMF model. We propose three different
energy difference functions and evaluate their impact on the performance of MAXMIN-
FAIR_SA on the 21 standard instances from track three of the ITC2007 competition
(McCollum et al. 2010). Our experiments indicate that the known best solutions with
respect to the CB-CTT model are quite good with respect to MMF, but further improve-
ments are possible for 20 out of 32 instances, which often result in a considerable benefit
for the worst-off stakeholders.

The fairness conditions imposed by MMF are rather strict in the sense that no trade-off
arises between the fairness of the penalty distribution and the sum of the penalties. In
practice however, it may be desirable to pick a timetable from a set of Pareto-optimal
solutions with respect to these two criteria. For this purpose, we propose the JFI-CB-CTT
problem, which is a bi-criteria optimization problem. We investigate the trade-offs
between fairness and efficiency for the six standard instances from (Di Gaspero and
Schaerf 2012) whose known best solutions have the highest total penalty of the whole set
of benchmark instances. Our choice of instances is motivated by the fact that if the total
penalty of a timetable is very small, then there is not much gain for anyone in distributing
the penalty in a fair way. Our conclusion regarding this approach is that, although the
known best solutions for the six instances are already quite fair, we can improve the fairness further at the cost of only a small increase of the total penalty. For a theoretical treatment of the price of fairness on so-called convex utility sets with respect to proportional fairness and MMF, see Bertsimas et al. (2011).

The remainder of this work is organized as follows. In Sect. 2, we will provide a brief review of the CB-CTT problem model as well as MMF and Jain’s fairness index (JFI). In Sect. 3 we will propose MMF-CB-CTT and JFI-CB-CTT, two fair variants of the CB-CTT problem formulation, and in Sect. 4, we will introduce the SA-based optimization algorithm MaxMinFair_SA for solving MMF-CB-CTT instances. Section 5 is dedicated to our experimental evaluation of the fairness of the known best solutions to 32 standard instances from the website (Di Gaspero and Schaerf 2012) with respect to MMF and JFI, and the performance of the MaxMinFair_SA algorithm.

2 Preliminaries

In this section, we provide a brief review of the CB-CTT problem formulation as well as relevant definitions concerning MMF and JFI.

2.1 The curriculum-based course timetabling problem

The CB-CTT is a particular variant of the UCTP. It has been proposed for the second international timetabling competition in 2007 (Di Gaspero et al. 2007), and has since then emerged as one of the standard problem formulations for benchmarking in the timetabling community. The central entities in the CB-CTT formulation are the curricula, which are sets of lectures that must not be taught simultaneously. Both problem formulations proposed in the next section are essentially the CB-CTT problem with a modified objective function.

The CB-CTT problem is NP-hard and a lot of effort has been devoted to the development of exact and heuristic methods which provide high quality solutions within reasonable time. A wide range of techniques has been employed for solving CB-CTT instances including but not limited to approaches based on Max-SAT (Asín Acha and Nieuwenhuis 2010), mathematical programming (Lach and Lübbeke 2010; Burke et al. 2011), local search (Di Gaspero and Schaerf 2006; Lü and Hao 2010), evolutionary computation (Abdullah et al. 2007) as well as hybrid approaches (Bellio et al. 2012; Müller 2009). There has been a lot of progress in terms of the achieved solution quality in recent years, see the website (Di Gaspero and Schaerf 2012) for current results.

A CB-CTT instance consists of a set of rooms, a set of courses, a set of curricula, a set of teachers and a set of days. Each day is divided into a fixed number of timeslots, a pair composed of a day and a timeslot is called a period. A period in conjunction with a room is called a resource. Each course consists of a number of lectures, i.e., a number of events to be scheduled, is taught by some teacher and has a number of students attending it. Each curriculum is a set of courses. For each room, we are provided with the maximum number of students it can accommodate and for each course, we are given a list of periods in which it must not be taught. A solution to a CB-CTT instance is a timetable, i.e., an assignment $\tau$ of courses to resources. The quality of a timetable is determined according to four hard and four soft constraints (Di Gaspero et al. 2007).

A timetable that satisfies all hard constraints is called feasible. Each violation of one of the soft constraints results in a penalty value for the timetable. The CT-CTT objective
function aggregates individual penalties by taking their weighted sum. Detailed descriptions of how hard and soft constraints are evaluated and how much penalty is applied for a particular soft constraint violation can be found in the report by Di Gaspero et al. (2007). In contrast, both proposed fair course timetabling problem formulations aggregate the penalties per stakeholder and evaluate the penalty distribution according to the fairness notion at hand. For the remainder of this paper, we will deal with feasible timetables exclusively, so there is no need to choose the relative importance soft constraint violations in the presence of hard constraint violations.

2.2 Fairness in resource allocation

Fairness issues typically arise when scarce resources are allocated to a number of stakeholders with demands. In computer science, fairness is an important aspect of the design and analysis of communication protocols (e.g., Bartal et al. 2002; Bertsekas and Gallager 1992; Bertsimas et al. 2011; Jain et al. 1984; Kelly et al. 1998; Kleinberg et al. 2001; Ogryczak and Wierzbicki 2004; Salles and Barria 2008). In operations research, fairness criteria have been applied, for example, to the aircraft landing problem (Soomer and Koole 2008) and the nurse rostering problem (Constantino et al. 2011; Smet et al. 2012).

Fairness in the academic setting has been considered, to some extent, in the generalized balanced academic curriculum (GBAC) problem proposed in (Di Gaspero and Schaerf 2008). The GBAC problem consists of allocating courses to teaching terms such that workload and prerequisite constraints are satisfied. The objective is to minimize the sum of the deviations from the average workload for each curriculum. The deviation from the average workload is essentially an inequality measure similar to the fairness index used in our proposed JFI-CB-CTT formulation. Since the sum of the deviations is minimized, a good solution may result in a small deviation from the average workload for some curricula at the expense of other curricula. Such situations are what we are trying to remedy with our proposed fair course timetabling problem formulations. For the GBAC problem however, some specifics limit these undesirable effects: there are fixed upper and lower limits for the workload per term and we can assume that the total workload for each curriculum is sensible.

Many resource allocation problems capture the task of maximizing the amount of resources allocated to the stakeholders or minimizing some quantity like cost, weight, or penalty. Fairness in this context means that the distribution of resources, costs and penalty over the stakeholders is important. A fairness model characterizes which distributions are preferable for a given problem. We will now discuss MMF and JFI, the two fairness models that will be used later on to create fair course timetables. Consider a resource allocation problem with \( n \) stakeholders. Each resource allocation (feasible solution) \( s \) (e.g., a timetable \( \tau \)) induces an allocation vector \( X = (x_1, \ldots, x_n) \) with non-negative rational entries, where each entry \( x_i \) corresponds to the amount of resources allocated to stakeholder \( i \), for \( 1 \leq i \leq n \).

2.2.1 Max–min fairness

MMF can be stated as iterated application of Rawls’s Second Principle of Justice (Rawls 1999):

Social and economic inequalities are to be arranged so that they are to be of greatest benefit to the least-advantaged members of society. (the Difference Principle)
Once the outcome for the least-advantaged group is as good as possible, the difference principle can be applied again to everyone except the least-advantaged group in order to achieve the best-possible outcome for the second least-advantaged group, and so on. The resulting allocation is called max–min fair. Due to the iterated application of the difference principle, a max–min fair allocation enforces an efficient resource usage to some extent. In particular, a max–min fair resource allocation is Pareto-optimal. To see this, assume that a max–min fair allocation \( X^* \) is dominated by an allocation \( X \). Then all stakeholders are at least as well-off with allocation \( X \) as they are with allocation \( X^* \) and there is at least one stakeholder whose outcome is better in allocation \( X \). Therefore \( X^* \) cannot be max–min fair.

We give a formalization of MMF that is similar to the one in (Kleinberg et al. 2001). In contrast to their bandwidth allocation setting we will later on deal with a max–min fair minimization problem and therefore the formalism needs to be adapted slightly. For a given allocation vector \( X \) we denote by \( \hat{X} \) the vector containing the entries of \( X \) arranged in non-increasing order. For allocation vectors \( X \) and \( Y \) we write \( X \preceq_{\text{mn}} Y \) if \( X \) is at least as good as \( Y \) in the max–min sense. For a max–min fair minimization problem, \( X \preceq_{\text{mn}} Y \) holds iff \( \hat{X} \preceq_{\text{lex}} \hat{Y} \), where \( \preceq_{\text{lex}} \) is the usual lexicographic comparison. Let \( s \) be a feasible solution to an instance \( I \) of a combinatorial minimization problem and let \( X \) be the allocation vector induced by \( s \). Then \( s \) is called max–min fair, if for any other solution \( s' \) to \( I \) we have \( X \preceq_{\text{mn}} Y \), where \( Y \) is the allocation vector induced by \( s' \). Since the lexicographic comparison is performed on the sorted allocation vectors, MMF does not discriminate between stakeholders. The comparison \( \preceq_{\text{mn}} \) is purely qualitative, that is, it does not quantify the fairness difference between two allocations.

A weaker version of MMF results if Rawl’s Second Principle of Justice is not applied iteratively, but just once. This means that we are concerned with choosing the best possible outcome for the worst-off stakeholder. In the literature, related optimization problems are referred to as bottleneck optimization problems (Edmonds and Fulkerson 1970; Punnen and Zhang 2011). Note that in contrast to MMF, an optimal solution to a bottleneck optimization problem is not necessarily Pareto-optimal. In the context of practical academic timetabling, the use of bottleneck optimization is hard to justify: each stakeholder is guaranteed to be at least as well off as the worst-off stakeholder, but no further improvement is considered.

2.2.2 Jain’s fairness index

Jain’s fairness index (JFI) is an inequality measure proposed by Jain et al. (1984), similar to the well-known Gini index (Gini 1921). Generally, a highly unequal distribution of resources is considered unfair. JFI is the crucial fairness measure that is used in the famous AIMD algorithm (Chiu and Jain 1989) used in TCP Congestion Avoidance. The fairness index \( J(X) \) of an allocation vector \( X \) is defined as follows:

\[
J(X) = \frac{\left( \sum_{1 \leq i \leq n} x_i \right)^2}{n \cdot \sum_{1 \leq i \leq n} x_i^2}.
\]

It has several useful properties like population size independence, scale independence (for all \( x \in \mathbb{R}^+ \), \( J(X) = J(ax) \)), it is bounded between 0 and 1, and it has an intuitive interpretation: if a total amount \( r \) of resources is distributed over \( n \) stakeholders and \( k < n \) stakeholders receive \( \frac{r}{k} \) resources while the remaining stakeholders receive nothing, then the fairness index is \( \frac{k}{n} \). In particular, \( J(X) = 1 \) means that all stakeholders receive the
same amount of resources in allocation \( X \), and \( J(X) = 1/n \) means that all resources are occupied by a single stakeholder, which is considered to be the most unfair allocation.

3 Fairness in academic course timetabling

In order to use the fairness models from the previous section, we need determine an allocation vector from a timetable. The central entities in the CB-CTT problem formulation are the curricula. Therefore, in this work, we are interested in a fair distribution of penalty over the curricula. Depending on the application, different stakeholders can be picked, but conceptually this does not change much. Let \( I \) be a CB-CTT instance with curricula \( c_1, c_2, \ldots, c_k \) and let \( f_c \) be the CB-CTT objective function from (Di Gaspero et al. 2007), which evaluates the four soft constraints restricted to the courses in curriculum \( c \). For a timetable \( \tau \) the corresponding allocation vector is given by the allocation function

\[
A(\tau) = (f_{c_1}(\tau), f_{c_2}(\tau), \ldots, f_{c_k}(\tau)).
\]

**Definition 1** (MMF-CB-CTT) Given a CB-CTT instance \( I \), the task is to find a feasible timetable \( \tau \) such that \( A(\tau) \) is max–min fair.

Any solution can be improved only if there is a curriculum \( c \) whose penalty can be reduced without increasing the penalty of another curriculum \( c' \) that receives at least as much penalty as \( c \).

In order to explore the trade-off between efficient and fair resource allocation in the context of the CB-CTT model, we propose another fair variant called JFI-CB-CTT that is based on JFI proposed by Jain et al. (1984). In order to get meaningful results from the fairness index however, we need a different allocation function. Consider an allocation \( X \) that distributes all penalty equally among \( r \) curricula, while the remaining \( k - r \) curricula receive no penalty. Then the intuitive interpretation of \( J(X) = r/k \) says that \( r \) curricula are happy with the allocation (Jain et al. 1984). In our situation however, the opposite is the case: \( k - r \) curricula are happy since they receive no penalty at all. The following allocation function shifts the penalty values such that the corresponding fairness index in the situation described above becomes \((k - r)/k\), which is in agreement with the intuitive interpretation of the fairness index:

\[
A'(\tau) = (f_{\text{max}} - f_{c_1}(\tau), f_{\text{max}} - f_{c_2}(\tau), \ldots, f_{\text{max}} - f_{c_k}(\tau)),
\]

where

\[
f_{\text{max}} = \max_{1 \leq i \leq k} \{f_{c_i}(\tau)\}.
\]

We let \( J(X) = 1 \) if \( X = (0, \ldots, 0) \), which covers the case \( r = k \) above.

**Definition 2** (JFI-CB-CTT) Given a CB-CTT instance \( I \), the task is to find the set of feasible solutions which are Pareto-optimal with respect to the two objectives of the objective function

\[
F(\tau) = (f(\tau), 1 - J(A'(\tau))),
\]

where \( f \) is the CB-CTT objective function from (Di Gaspero et al. 2007) and \( J \) is from Eq. (1).
As suggested in (Mühlenthaler and Wanka 2012), fairness conditions can be added in an analogous manner to other variants of educational timetabling problems, such as the post-enrollment course timetabling problem and the exam timetabling problem, as well as the nurse rostering problem. For example, the central entities of interest in the post-enrollment course timetabling problem are the individual students. Therefore, the goal were to achieve a fair distribution of penalty over all students. Once an appropriate allocation function has been defined, we immediately get the corresponding optimization problem with a fairness objective. Similar approaches have been explored for the nurse rostering problem (Smet et al. 2012; Martin et al. 2013) and the examination timetabling problem (Muklason et al. 2013).

Our proposed problem formulations are concerned with balancing the interests between stakeholders, who are in our case the students. In practice however, there are often several groups of stakeholders with possibly conflicting interest, for example students, lecturers and administration. There are several possible ways for extending the problem formulations above to include multiple stakeholders. For example, a multi-objective optimization approach may be considered, where each objective captures the fairness with respect to a particular stakeholder. When using inequality measures like JFI for different groups of stakeholders, the inequality values can be aggregated, for instance using a weighted-sum or ordered weighted averaging (Yager 1988) approach. Furthermore, MMF or a suitable inequality measure can be applied to the different objectives to balance the interests of the different groups of stakeholders.

4 Simulated annealing for the MMF-CB-CTT problem

SA is a popular local search method proposed by Kirkpatrick et al. (1983), which works surprisingly well on many problem domains. SA has been applied successfully to timetabling problems (e.g., see Kostuch 2004; Thompson and Dowsland 1996). Some of the currently known best solutions to CB-CTT instances from the ITC2007 competition were discovered by SA-based methods according to the website (Di Gaspero and Schaerf 2012). The SA algorithm generates a new candidate solution according to some neighborhood exploration method, and replaces the current solution with a certain probability depending on (i) the quality difference between the two solutions and (ii) the current temperature. Since MMF only tells us which of two given solutions is better, but not how much better, the main challenge in tailoring SA to max–min fair optimization problems is to find a suitable energy difference function, which quantifies the difference in quality between two candidate solutions. In the following, we propose three different energy difference measures for max–min fair optimization and provide details on the acceptance criterion, the cooling schedule, and the neighborhood exploration method chosen for the experimental evaluation of our approach in the next section. We refer to our adaptation of SA to the MMF-CB-CTT problem as MAXMINFAIR_SA.

4.1 Acceptance probability

Similar to the original SA algorithm, a new solution \( \tau_{\text{next}} \) is accepted if it is at least as good as the current solution \( \tau_{\text{cur}} \), i.e., \( A(\tau_{\text{next}}) \leq \text{mm} A(\tau_{\text{cur}}) \). If \( \tau_{\text{next}} \) is worse than \( \tau_{\text{cur}} \) then it is accepted with probability...
\[
\exp\left(-\frac{\Delta E(A(\tau_{\text{cur}}), A(\tau_{\text{next}})))}{\delta}\right),
\]

where \(\Delta E\) is the “energy difference” between the allocations \(A(\tau_{\text{cur}})\) and \(A(\tau_{\text{next}})\). The energy difference essentially quantifies how much worse one allocation is compared to another allocation. We propose three different functions for measuring the energy difference: \(\Delta E_{\text{lex}}, \Delta E_{\text{cw}}\), and \(\Delta E_{\text{ps}}, \Delta E_{\text{lex}}\) derives the energy difference from a lexicographic comparison, \(\Delta E_{\text{cw}}\) from the component-wise ratios of the sorted allocation vectors and \(\Delta E_{\text{ps}}\) from the ratios of the partial sums of the sorted allocation vectors. Our experiments presented in the next section indicate that the choice of the energy difference measure has a clear impact on the performance of \textsc{MaxMinFair-SA} and is thus a critical design choice.

For an allocation vector \(X\), let \(\tilde{X}_i\) denote the \(i\)th entry after sorting the entries of \(X\) in non-increasing order. The energy difference \(\Delta E_{\text{lex}}\) of two allocation vectors \(X\) and \(Y\) of length \(n\) is defined as follows:

\[
\Delta E_{\text{lex}}(X, Y) = 1 - \frac{1}{n} \cdot \min_{1 \leq i \leq n} \left\{ i \mid \tilde{Y}_i > \tilde{X}_i \right\} - 1
\]

\(\Delta E_{\text{lex}}\) determines the energy difference between \(X\) and \(Y\) from the index that determines that \(X \succeq_{\text{min}} Y\). Thus, sorted allocation vectors which differ at the most significant indices have a higher energy difference than those differing at later indices. In particular, \(\Delta E_{\text{lex}}\) evaluates to 1 if the minimum is 1, and it evaluates to \(1/n\) if the minimum is \(n\).

The energy difference measure \(\Delta E_{\text{lex}}\) considers the earliest index at which two sorted allocation vectors differ but not how much the entries differ. We additionally propose the two energy difference measures \(\Delta E_{\text{cs}}\) and \(\Delta E_{\text{ps}}\) which take this information into account. These energy difference measures are derived from the definitions of approximation ratios for max–min fair allocation problems given in (Kleinberg et al. 2001). An approximation ratio is a measure for how much worse the quality of a solution is relative to a possibly unknown optimal solution. In our case, we are interested in how much worse one given allocation is relative to another given allocation. We need to introduce some modifications of the definitions in (Kleinberg et al. 2001) since we are dealing with a minimization problem.

Note that due to (2), an allocation vector does not contain any negative entries. Let \(\mu_{X,Y}\) be the largest value of the two allocation vectors \(X\) and \(Y\) offset by a parameter \(\delta > 0\), i. e.,

\[
\mu_{X,Y} = \max\{\tilde{X}_1, \tilde{Y}_1\} + \delta.
\]

The offset \(\delta\) is introduced in order to avoid divisions by zero when taking ratios of penalty values.

The component-wise energy difference \(\Delta E_{\text{cw}}\) of allocation vectors \(X\) and \(Y\) is defined as follows:

\[
\Delta E_{\text{cw}}(X, Y) = \max_{1 \leq i \leq n} \left\{ \frac{\mu_{X,Y} - \tilde{Y}_i}{\mu_{X,Y} - \tilde{X}_i} \right\} - 1
\]

Since all entries are subtracted from \(\mu_{X,Y}\), the ratios of the most significant entries with respect to \(\succeq_{\text{min}}\) tend to dominate the value of \(\Delta E_{\text{cw}}\). Consider for example the situation that \(Y\) is much less fair than \(X\), say, \(\max\{\tilde{X}_1, \tilde{Y}_1\}\) occurs more often in \(X\) than in \(Y\). Then for a
small offset \( \delta \) the energy difference \( \Delta E_{cw}(X, Y) \) becomes large. On the other hand, if \( X \) is nearly as fair as \( Y \) then the ratios are all close to one and thus \( \Delta E_{cw}(X, Y) \) is close to zero.

The third proposed energy difference measure \( \Delta E_{ps} \) is based on the ratios of the partial sums \( \sigma_i(X) \) of the sorted allocation vectors.

\[
\sigma_i(X) = \sum_{1 \leq j \leq i} \tilde{X}_j.
\]

The intention of using partial sums of the sorted allocations is to give the stakeholders receiving the most penalty more influence on the resulting energy difference compared to \( \Delta E_{cw} \). The energy difference \( \Delta E_{ps} \) is defined as:

\[
\Delta E_{ps}(X, Y) = \max_{1 \leq i \leq n} \left\{ \frac{i \cdot \mu_{X,Y} - \sigma_i(\tilde{Y})}{i \cdot \mu_{X,Y} - \sigma_i(\tilde{X})} \right\} - 1.
\] (8)

4.2 Cooling schedule

The temperature at time \( t \) is determined according to a standard geometric cooling schedule:

\[
\vartheta(t) = \vartheta' \cdot \vartheta_{max},
\]

where \( \vartheta \) is the cooling rate. Geometric cooling schedules are widely used in practice and are known to work well in many problem domains including timetabling problems (e.g., see Laarhoven and Aarts 1987; Koulamas et al. 1994; Thompson and Dowsland 1998). In order to make the algorithm behavior more consistent for different timeouts, instead of specifying the cooling rate, we determine it from \( \vartheta_{max} \), the desired minimum temperature \( \vartheta_{min} \) and the timeout according to:

\[
\vartheta = \left( \frac{\vartheta_{min}}{\vartheta_{max}} \right)^{-\frac{1}{\text{timeout}}}. \tag{9}
\]

Thus, at the beginning (\( t = 0 \)) the temperature level is \( \vartheta_{max} \) and when the timeout is reached (\( t = \text{timeout} \)), the temperature level becomes \( \vartheta_{min} \). We chose to set a timeout rather than a maximum number of iterations since this setting is compliant with the ITC2007 competition conditions, which are a widely accepted standard for comparing results.

4.3 Neighborhood

In each step, \textsc{MaxMinFair-SA} picks at random a neighbor of the current solution from the Kempe-neighborhood. The Kempe-neighborhood is the set of all timetables which can be reached by performing a single Kempe-move such that the number of lectures per period do not exceed the number of available rooms. The Kempe-move is a well-known and widely used operation for swapping events in a timetable (e.g., see Burke et al. 2010; Lü and Hao 2010; Merlot et al. 2002; Thompson and Dowsland 1998; Tuga et al. 2007). A prominent feature of the Kempe-neighborhood is that it contains only moves that preserve the feasibility of a timetable. Since the algorithm \textsc{MaxMinFair-SA} only uses moves from this neighborhood the output is guaranteed to be a feasible timetable. In the future, more advanced neighborhood exploration methods similar to the approaches in (Di Gaspero and
Schaerf 2006; Lü and Hao 2010) could be used, which may well lead to an improved overall performance of MAXMINFAIR_SA.

5 Evaluation

We will first address the question how fair the known best timetables for 32 CB-CTT instances are with respect to JFI and MMF. We will then compare the results to timetables obtained using the MMF-CB-CTT and JFI-CB-CTT problem formulations. Table 1 shows our evaluation of the known best solutions to the instances comp01, comp02, ..., comp21 from the ITC2007 competition, as well as seven additional instances from the University of Udine, DDS1, ..., DDS7, and four large instances from the University of Erlangen (erlangen-2011-2, erlangen-2012-1, erlangen-2012-2, and erlangen-2013-1). All instances are available from the website (Di Gaspero and Schaerf 2012). Please note that the known best timetables were not created with fairness in mind, but the objective was to create timetables with minimal total penalty according to the CB-CTT objective function.

In Table 1, $s_{best}$ is the known best solution for the given instance. $A$ and $A'$ refer to the allocation functions given in (2) and (3), respectively. The data indicates that, according to JFI, the timetables with a low total penalty are also rather fair. This can be explained by the fact that these timetables do not have a large amount of penalty to distribute over the curricula. Thus, most curricula receive little or no penalty and consequently, the distribution is fair for most curricula. Note that $J(X) = 1$ if all entries of $X$ are zero, as defined in Sect. 3 We will show in Sect. 5.2, that for solutions to the ITC2007 instances with rather large total penalty there is still some room for improvement with respect to the fairness index $J$. The rightmost column of Table 1 shows the distribution of penalty over the curricula according to the MMF-CB-CTT objective function. For the purpose of succinctness, the different penalty values are shown in the bases and the exponents denote the multiplicity of the penalty values. For example, the allocation vector (5, 5, 0, 0, 0) is represented as $5^20^3$. The sum of the values of an allocation vector is generally much larger than the CB-CTT objective value. The reason for this is that the penalty related to a course is counted for each curriculum the course belongs to. With a few exceptions the overall impression is that the penalty is assigned to only a few curricula while a majority of curricula receives no penalty. In the next section we will show that an improvement is possible for the worst-off curricula by using the MMF-CB-CTT model.

5.1 Max–min fair optimization

In Sect. 4, we presented algorithm MAXMINFAIR_SA for solving max–min fair minimization problems. A crucial part of this algorithm is the energy difference measure which determines how much worse a given solution is compared to another solution, i.e., the energy difference of the solutions. We evaluate the impact of the three energy difference measures (5), (7) and (8) on the performance of MAXMINFAIR_SA.

Our setup was the following: For each energy difference function we independently performed 50 runs of MAXMINFAIR_SA. The temperature levels were determined experimentally, we set $\theta_{max} = 5$ and $\theta_{min} = 0.01$; the cooling rate $\alpha$ was set according to Eq. (9). In order to establish consistent experimental conditions for fair optimization, we used a timeout, which was determined according to the publicly available ITC2007 benchmark executable. On our machines (i7 CPUs running at 3.4 GHz, 8 GB RAM), the timeout was
set to 192 s. The MaxMinFair_SA algorithm was executed on a single core. We generated feasible initial timetables for MaxMinFair_SA as a preprocess using sequential heuristics proposed in (Burke et al. 2007). The soft constraint violations were not considered at this stage. Since the preprocess was performed only once per instance (not per run), it is not counted in the timeout. The time taken by the preprocessing was less than 1 s per instance, which is negligible compared to the timeout.

Table 2 shows the impact of the parameter $\delta$ on the performance of MaxMinFair_SA with energy difference measure $\Delta E_{cw}$. We performed the one-sided Wilcoxon Rank-Sum test (see Wilcoxon 1945; Lapin 1990) with a significance level of 0.01 to determine if the...
Parameter $\delta$ has significant performance implications in this setting. For each combination $(\delta_1, \delta_2)$ the table shows on which instances $\delta = \delta_1$ outperformed $\delta = \delta_2$.

| $\delta_1$ | $\delta_2$ | $10^0$ | $10^{-2}$ | $10^{-3}$ | $10^{-6}$ |
|------------|------------|-------|-----------|-----------|-----------|
| $10^0$     | –          | 02    | 02, 05    | 02        |           |
| $10^{-2}$  | 10, 19, 20 | –     | 09        | 19        |           |
| $10^{-3}$  | 01, 10, 19, 20 | –     | –         | 03, 19    |           |
| $10^{-6}$  | 01, 10, 20  | –     | –         | –         |           |

We further investigated the impact of the three energy difference measures $\Delta E_{\text{lex}}, \Delta E_{\text{cw}}$ and $\Delta E_{\text{ps}}$ on the performance of $\text{MaxMinFair-SA}$. Like above, we used the Wilcoxon Rank-Sum test with a significance level of 0.01. Table 3 shows for any two energy difference measures $E_1, E_2 \in \{\Delta E_{\text{lex}}, \Delta E_{\text{cw}}, \Delta E_{\text{ps}}\}$ the ITC2007 instances for which setting $\Delta E = E_1$ is better than setting $\Delta E = E_2$. The data clearly shows that $\Delta E_{\text{cw}}$ is the best choice among the three alternatives, since it is a better choice than $\Delta E_{\text{lex}}$ on all instances except $\text{comp11}$ and a better choice than $\Delta E_{\text{ps}}$ on $\text{comp06}, \text{comp07}, \text{comp08}, \text{comp17}$ and $\text{comp21}$. It is only beaten by $\Delta E_{\text{ps}}$ on $\text{comp18}$.

The data in Table 4 shows a comparison of the sorted allocation vectors of the known best solutions to 32 CB-CTT instances from the website (Di Gaspero and Schaerf 2012) with the best solutions produced by $\text{MaxMinFair-SA}$ with $\Delta E = \Delta E_{\text{cw}}$. First of all, for instances $\text{comp01}, \text{comp11}, \text{DDS2}, \text{DDS3}$, and $\text{DDS5}$, the allocation vectors of the best existing solutions and the best solution found by $\text{MaxMinFair-SA}$ are identical. For these instances the known best solutions are optimal and have no soft constraint violations. Thus, these solutions are also optimal with respect to MMF. This means that $\text{MaxMinFair-SA}$ finds optimal solutions for five out of 32 instances. We can also conclude from the data shown in Table 4, that the maximum penalty any curriculum receives is significantly less for most instances and the penalty is more evenly distributed across the curricula. This means that although max–min fair timetables may have a higher total penalty, they might be more attractive from the students’ perspective, since in the first place each student notices an unfortunate arrangement of his/her timetable, which is tied to the curriculum. Furthermore, we can observe that if the total penalty of a known best solution is rather low, then it is also good with respect to MMF. For several instances in this category, $(\text{comp02}, \text{comp07}, \text{comp10}, \text{comp20}, \text{DDS1}, \text{and DDS6})$, the solution found by $\text{MaxMinFair-SA}$ is not as good as the known best solution with respect to MMF. We can conclude that if there is not much penalty to distribute between the stakeholders, it is not necessary to enforce a fair distribution of penalty.

We compared the best solutions from (Di Gaspero and Schaerf 2012) to the best solutions produced by $\text{MaxMinFair-SA}$ as shown in Table 4 with respect to the original CB-CTT objective function. On average, for the 21 instances from the competition, the total penalty increases by a factor of 3.7. However, this is mostly due to instances with very
low total penalty, such as \texttt{comp07}, \texttt{comp10} and \texttt{comp20}, where \texttt{MAXMINFAIR-SA} was also not able to beat the known best solutions with respect to MMF. On 10 instances, the total penalty of the best solutions found by \texttt{MAXMINFAIR-SA} was less than twice the penalty of the known best solutions. If the increase of the overall penalty is considered too high or the benefits for the worst-off stakeholders are not significant, it may be desirable to tune the tradeoff between fairness and overall penalty, which is possible with the JFI-CB-CTT model.

5.2 The trade-off between fairness and efficiency

We proposed the JFI-CB-CTT problem formulation in Sect. 3, which allows us to investigate the trade-off between fairness and efficiency which arises in course timetabling. Concerning the fairness indices shown in Table 1, we can observe that for all of the CB-CTT instances the fairness index is greater than 0.8. In order to solve the corresponding JFI-CB-CTT instances, we use the multi-objective optimization algorithm AMOSA (Bandyopadhyay et al. 2008) that is based on SA like Algorithm \texttt{MAXMINFAIR-SA}. Since we do not expect from a general multi-objective optimization algorithm to produce solutions as good as the best CB-CTT solvers, we will consider the following scenario to explore the trade-offs between fairness and efficiency: starting from the known best solution we examine how much increase in total penalty we have to tolerate in order to increase the fairness further. We will take as examples the six instances from the ITC2007, whose known best solutions have the highest total penalty, i.e., \texttt{comp03}, \texttt{comp05}, \texttt{comp09}, \texttt{comp12}, \texttt{comp15} and \texttt{comp21}.

The temperature levels for the AMOSA algorithm were set to $\theta_{\text{max}} = 20$ and $\theta_{\text{min}} = 0.01$; $\alpha$ was set according to Eq. (9) with a timeout determined by the official ITC2007 benchmark executable. The plots in Figure 1 show the (Pareto-) non-dominated solutions found by AMOSA. The arrows point to the starting point, i.e., the best available solutions to the corresponding instances. For instances \texttt{comp05} and \texttt{comp21} solutions with a lower total cost than the the previously known best solutions were discovered by this approach\footnote{Very recently, Antony Philips managed to further improve the total penalty by one for \texttt{comp21}. This solution matches the known lower bound and is thus optimal, see (Di Gaspero and Schaerf 2012).}. The plots show that the price for increasing the fairness is generally not very high—up to a certain level, which depends on the instance. In fact, for \texttt{comp09} and \texttt{comp21}, the fairness index can be increased by 3.5 and 1.4 \%, respectively, without increasing the total penalty at all.

In Figure 1, the straight lines that go through the initial solutions show a possible trade-off between fairness and efficiency: the slopes were determined such that a 1 \% increase in
Table 4  Allocation vectors of the current best solutions from (Di Gaspero and Schaerf 2012) and the best solutions found by MaxMinFair-SA

| Instance       | Known best solution | MaxMinFair-SA ($\Delta E = \Delta E_{\text{w}}$) |
|----------------|---------------------|-----------------------------------------------|
| comp01         | $5^2, 0^{12}$       | $5^2, 0^{12}$                                 |
| comp02         | $4, 2^{10}, 0^{59}$ | $4^2, 3^{11}, 1^7, 0^{30}$                    |
| comp03         | 13, 10^3, 9, 7^2, 6^4, 5^{13}, 4, 2^6, 0^{17} | $6^4, 4^{11}, 2^{22}, 1^3, 0^{28}$            |
| comp04         | 7, 6^3, 5^4, 2, 0^{46} | $6^4, 4^2, 2^1, 1^0, 0^{46}$                  |
| comp05         | 41^2, 36^2, 35^5, 32^5, 31^6, 30^9, 28, 27, 0^{60} | $19^2, 18^3, 17^3, 16^5, 15^2, 14^{15}, 13^8, 12^4, 11^{13}, 10^24$ |
| comp06         | 12, 7^2, 5^4, 2^3, 0^{60} | 6, 2^{23}, 1^{24}, 0^{29}                     |
| comp07         | $6, 0^{76}$          | $6^4, 4^2, 2^1, 1^5, 0^{43}$                  |
| comp08         | 7, 6^3, 5^4, 2^2, 0^{69} | $6^4, 4^{14}, 2^{17}, 0^{38}$                 |
| comp09         | 10^5, 9, 7^{10}, 6^6, 5^{10}, 4, 2, 0^{41} | $2^{19}, 1^{6}, 0^{42}$                      |
| comp10         | $2^2, 0^{65}$        | $1^{03}$                                      |
| comp11         | $0^{11}$             |                                               |
| comp12         | 45, 30^{14}, 28, 27^2, 26^5, 25^{19}, 22^{24}, 21^{26}, 20^{30}, 19^{34}, ... | $10^3, 9^6, 8^{31}, 7^7, 6^{45}, 5^2, 4^{36}, ...$ |
| comp13         | 8, 7, 6^5, 5^4, 4^2, 2^3, 0^{47} | $6^6, 4^4, 2^{13}, 1^6, 0^{39}$               |
| comp14         | $8^4, 7, 5^6, 2^6, 0^47$ | $8^4, 4^2, 3, 2^{18}, 0^{38}$                 |
| comp15         | $10^5, 9^3, 7, 6^4, 5^{13}, 4, 2^7, 0^{36}$ | $6^4, 4^{14}, 2^{23}, 1^2, 0^28$              |
| comp16         | $7^2, 5^7, 4, 0^{61}$ | $4^5, 2^{16}, 1^4, 0^{46}$                    |
| comp17         | $10^5, 6^3, 5^8, 2^4, 0^{52}$ | $10^2, 6^2, 4^7, 3, 2^{25}, 1^7, 0^{26}$     |
| comp18         | $17, 15, 14, 13, 11, 10, 9^2, 5^{19}, 2^2, 0^{23}$ | $4^{20}, 2^{11}, 1^3, 0^{16}$                 |
| comp19         | $13, 7, 6^4, 5^2, 4, 2^7, 0^{30}$ | $6^4, 4^6, 2^{15}, 1^{14}, 0^{27}$           |
| comp20         | $2^2, 0^{66}$        | $4^5, 3^3, 2^{31}, 1^7, 0^{12}$               |
| comp21         | 12, 11, 10^7, 9, 7^4, 6^4, 5^{12}, 4, 2^3, 0^{47} | $10, 6^4, 5, 4^{15}, 3, 2^{36}, 1^3, 0^{17}$ |
| DDS1           | $5^{21}, 2^{9}, 0^{69}$ | $6^4, 5^{11}, 4^{17}, 3^3, 2^{22}, 1^{17}, 0^{25}$ |
| DDS2           | $0^{11}$             | $0^{11}$                                      |
| DDS3           | $0^9$                | $0^9$                                         |
| DDS4           | $1^{53}, 2, 0^{101}$ | $13^3, 11, 9^3, 7, 6^2, 4^5, 3^{11}, 2^{37}, 1^{25}, 0^{17}$ |
| DDS5           | $0^{44}$             | $0^{44}$                                      |
| DDS6           | $0^{62}$             | $2^{10}, 1^{11}, 0^{41}$                     |
| DDS7           | $0^{37}$             | $0^{37}$                                      |
| erlangen-2011-2| 20, 19^2, 17^5, 16, 15^{14}, 14^8, 12^{19}, ... | $16, 15, 14^8, 13^2, 12^{24}, 11^4, 10^97, ...$ |
| erlangen-2012-1| 22, 20^5, 17, 15^4, 12^{10}, 10^{19}, 9^8, 8, ... | $10^{28}, 9^2, 8^{34}, 7^2, 6^{253}, 5^2, 4^{222}, ...$ |
| erlangen-2012-2| 56, 53^3, 51^10, 49^8, 48, 46, 44^7, 19^2, ... | $52^7, 50^9, 48^{10}, 46^4, 44, 23, 21, 20, 19^{13}, ...$ |
| erlangen-2013-1| 19, 17, 15^4, 14^2, 12^6, 11^3, 10^6, 9^8, ... | $12^{18}, 10^{69}, 8^{156}, 7, 6^{264}, 4^{145}, 2^{75}, 0^{10}$ |

The allocation vectors marked in bold face indicate whether the known best solution or MaxMinFair-SA is better for a given instance.

Fairness yields a 1% increase in penalty. For the instances shown in Figure 1, the solutions remain close to the trade-off lines up to a fairness of 94–97%, while a further increase in fairness demands a significant increase in total cost. For the instances comp05, comp09 and comp15, there are several solutions below the trade-off lines. Picking any of the solutions below these lines would result in an increased fairness without an equally large
increase in the amount of penalty. This means picking a fairer solution might well be an attractive option in a real-world application. For comp05 for example, the fairness of the formerly best known solution with a total penalty of 291 can be increased by 5.4 % at 302 total penalty, which is a 3.8 % increase.

In summary, improving the fairness of a timetable that has been optimized for a low overall penalty as a post-processing step seems to be a viable approach for practical

Fig. 1 Non-dominated solutions found by the AMOSA algorithm for the JFI-CB-CTT versions of instances comp03, comp05, comp09, comp12, comp15 and comp21. All graphs show the fairness index on the horizontal axis and the amount of penalty on the vertical axis.
decision making. This means we can use proven existing algorithms to generate a good starting point, explore fairer alternatives and then choose the desired trade-off.

6 Conclusion

We introduced two problem formulations based on the CB-CTT problem, MMF-CB-CTT and JFI-CB-CTT. Both formulations include fairness criteria in the objective function and differ with respect to the fairness model they employ. Fairness in our setting means that the penalty assigned to a timetable is distributed in a fair way among the different curricula. The MMF-CB-CTT formulation aims at creating max–min fair course timetables while JFI-CB-CTT is a bi-objective problem formulation based on JFI. The motivation for the JFI-CB-CTT formulation is to explore the trade-offs between a fair penalty distribution and a low total penalty, which is not possible when using MMF criteria.

Furthermore we proposed the algorithm MAXMINFAIR_SA, a variant of SA tailored to the MMF-CB-CTT problem. A critical part of the algorithm is concerned with measuring the energy difference between two timetables, that is, how much worse a timetable is compared to another timetable with respect to MMF. Our experiments indicated that the choice of the energy difference measure has a significant impact on the performance of MAXMINFAIR_SA. Our results show that one energy difference measure that performs best on our set of 32 benchmark instances.

We investigated the fairness of the known best solutions of the 32 CB-CTT instances with respect to MMF and JFI. These solutions were not created with fairness in mind, but our results show that the solutions are generally quite fair. Nevertheless, we showed that some improvements are possible with respect to both MMF and JFI. The timetables produced by our proposed MAXMINFAIR_SA algorithms are better than the known best ones with respect to MMF for 20 out of 32 instances. Our investigation of the trade-off between fairness and the total amount of penalty using the JFI-CB-CTT problem formulation shows that the fairness of the known best timetables can be increased further with only a small increase of the total penalty.

Acknowledgements We would like to thank the anonymous referees for their constructive comments and valuable remarks on this paper. Research funded in parts by the School of Engineering of the University of Erlangen-Nuremberg.

References

Abdullah, S., Burke, E. K., & McCollum, B. (2007). A hybrid evolutionary approach to the university course timetabling problem. In IEEE congress on evolutionary computation (CEC) (pp. 1764–1768). doi:10.1109/CEC.2007.4424686.

Asín Acha, R., & Nieuwenhuis, R. (2010). Curriculum-based course timetabling with SAT and MaxSAT. In Proceedings of the 8th international conference on the practice and theory of automated timetabling (PATAT) (pp. 42–56).

Bandyopadhyay, S., Saha, S., Maulik, U., & Deb, K. (2008). A simulated annealing-based multiobjective optimization algorithm: AMOSA. IEEE Transactions on Evolutionary Computation, 12, 269–283. doi:10.1109/TEVC.2007.900837.

Bartal, Y., Farach-Colton, M., Yoospeh, S., & Zhang, L. (2002). Fast, fair and frugal bandwidth allocation in ATM networks. Algorithmica, 33(3), 272–286. doi:10.1007/s00453-001-0119-2.

Bellio, R., DiGaspero, L., & Schaerf, A. (2012). Design and statistical analysis of a hybrid local search algorithm for course timetabling. Journal of Scheduling, 15, 49–61. doi:10.1007/s10951-011-0224-2.

Bertsekas, D. P., & Gallager, R. (1992). Data networks (2nd ed.). Upper Saddle River: Prentice Hall.
Martin, S., Ouelhadj, D., Smet, P., & Vanden Berghe, G., & Özcan, E. (2013). Cooperative search for fair nurse rosters. *Expert Systems with Applications, 40*(16), 6674–6683. doi:10.1016/j.eswa.2013.06.019.

McCollum, B., Schaerf, A., Paechter, B., McMullan, P., Lewis, R., Parkes, A. J., Di Gaspero, L., Qu, R., & Burke, E. K. (2010). Setting the research agenda in automated timetabling: The second international timetabling competition. *INFORMS Journal on Computing, 22*, 120–130. doi:10.1287/ijoc.1090.0320.

Merlot, L. T. G., Boland, N., Hughes, B. D., & Stuckey, P. J. (2002). A hybrid algorithm for the examination timetabling problem. In *Proceedings of the 4th international conference on the practice and theory of automated timetabling (PATAT)* (pp. 207–231). doi:10.1007/3-540-45157-0_14.

Mühlenthaler, M., & Wanka, R. (2012). Fairness in academic timetabling. In *Proceedings of the 9th international conference on the practice and theory of automated timetabling (PATAT)* (pp. 114–130).

Muklason, A., Parkes, A. J., McCollum, B., & Özcan, E. (2013). Initial results on fairness in examination timetabling. In *Proceedings of the 6th multidisciplinary international conference on scheduling: Theory and applications (MISTA)* (pp. 777–780).

Müller, T. (2009). ITC2007 solver description: A hybrid approach. *Annals of Operations Research, 172*(1), 429–446. doi:10.1007/s10479-009-0644-y.

Ogryczak, W. (2010). Bicriteria models for fair and efficient resource allocation. In *Proceedings of the 2nd international conference on social informatics (SoCInfo)* (pp. 140–159). doi:10.1007/978-3-642-16567-2_11.

Ogryczak, W., & Wierzbicki, A. (2004). On multi-criteria approaches to bandwidth allocation. *Control and Cybernetics, 33*, 427–448.

Punnen, A. P., & Zhang, R. (2011). Quadratic bottleneck problems. *Naval Research Logistics (NRL), 58*(2), 153–164. doi:10.1002/nav.20446.

Rawls, J. (1999). *A theory of justice, revised edn*. Cambridge: Belknap Press of Harvard University Press.

Soomer, M. J., & Koole, G. M. (2008). Fairness in the aircraft landing problem. In *Proceedings of the Anna Valicek competition 2008*.

Soomer, M. J., & Koole, G. M. (2008). Fairness in the aircraft landing problem. In *Proceedings of the Anna Valicek competition 2008*.

Thompson, J., & Dowsland, K. A. (1996). General cooling schedules for a simulated annealing based timetabling system. In *Proceedings of the 1st international conference on the practice and theory of automated timetabling (PATAT)* (pp. 345–363). doi:10.1007/3-540-61794-9_70.

Thompson, J. M., & Dowsland, K. A. (1998). A robust simulated annealing based examination timetabling system. *Computers & Operations Research, 25*(7–8), 637–648. doi:10.1016/S0305-0548(97)00101-9.

Tuga, M., Berretta, R., & Mendes, A. (2007). A hybrid simulated annealing with Kempe chain neighborhood for the university timetabling problem. In *Proceedings of the 6th ACIS international conference on computer and information science (ACIS-ICIS)* (pp. 400–405). doi:10.1109/ICIS.2007.25.

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin, 1*(6), 80–83.

Yager, R. R. (1988). On ordered weighted averaging aggregation operators in multicriteria decisionmaking. *IEEE Transactions on Systems, Man and Cybernetics, 18*(1), 183–190, doi:10.1109/21.87068.