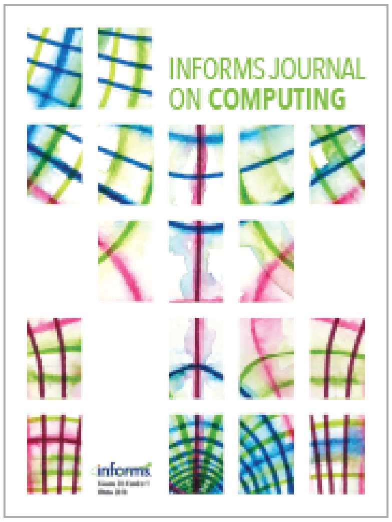


INFORMS Journal on Computing

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Optimization-based Mechanisms for the Course Allocation Problem

Hoda Atef Yekta, Robert Day

INFORMS JOURNAL ON COMPUTING
Vol. 32, No. 3, Summer 2020, pp. 641–660
ISSN 1091-9856 (print), ISSN 1526-5528 (online)
http://pubsonline.informs.org/journal/ijoc

Optimization-based Mechanisms for the Course Allocation Problem

Hoda Atef Yekta,[a] Robert Day[b]

[a] Department of Management, Youngstown State University, Youngstown, Ohio 44555; [b] Department of Operations and Information Management, University of Connecticut, Storrs, Connecticut 06269

Contact: hatefyekta@ysu.edu (HAY); bob.day@business.uconn.edu, http://orcid.org/0000-0002-9705-4130 (RD)

Received: February 27, 2016
Revised: February 17, 2017; October 2, 2017; July 10, 2018
Accepted: July 16, 2018
Published Online in Articles in Advance: January 13, 2020

https://doi.org/10.1287/ijoc.2018.0849

Abstract. In recent years, several universities have adopted an algorithmic approach to the allocation of seats in courses, for which students place bids (typically by ordering or scoring desirable courses), and then seats are awarded according to a predetermined procedure or mechanism. Designing the appropriate mechanism for translating bids into student schedules has received attention in the literature, but there is currently no consensus on the best mechanism in practice. In this paper, we introduce five new algorithms for this *course-allocation problem*, using various combinations of matching algorithms, second-price concepts, and optimization, and compare our new methods with the natural benchmarks from the literature: the (proxy) draft mechanism and the (greedy) bidding-point mechanism. Using simulation, we compare the algorithms on metrics of fairness, efficiency, and incentive compatibility, measuring their ability to encourage truth telling among boundedly rational agents. We find good results for all of our methods and that a two-stage, full-market optimization performs best in measures of fairness and efficiency but with slightly worse incentives to act strategically compared with the best of the mechanisms. We also find generally negative results for the bidding-point mechanism, which performs poorly in all categories. These results can help guide the decision of selecting a mechanism for course allocation or for similar assignment problems, such as project team assignments or sports drafts, for example, in which efficiency and fairness are of utmost importance but incentives must also be considered. Additional robustness checks and comparisons are provided in the online supplement.

History: Accepted by David Woodruff, Area Editor for Software Tools.
Supplemental Material: The online supplement is available at https://doi.org/10.1287/ijoc.2018.0849.

Keywords: computational economics • games • group decisions • bidding auctions

1. Introduction

The allocation of scarce classroom capacity to students is a challenging process for many universities when several particular courses experience significant excess demand. Similarities in students' preferences, limits on the number of semesters to complete a program, and a limited number of seats (especially in MBA programs) intensify the importance of what has become known as the *course-allocation problem* (CAP). In the last three decades, several universities, especially highly ranked (and, thus, heavily demanded) business schools, such as Stanford Business School, Harvard Business School, Ross School of Business at University of Michigan, and Columbia Business School, have tried to improve their mechanisms for allocating courses among students and have tested different methods to manage and solve this problem (Krishna and Ünver 2008).

In this paper, we consider several metrics relevant to a university in making these decisions to compare the practicality of newly introduced optimization-based approaches and two standard benchmarks on simulated data. These metrics can be categorized into three groups and described as follows:

Measures of Efficiency

An allocation is *Pareto efficient* if no student can be made better off without making another student worse off. Some mechanisms are not even Pareto efficient with respect to the submitted preferences (i.e., may not be Pareto efficient even when all players reveal their preferences truthfully).

The *allocative efficiency* of a market outcome measures the total market utility (satisfaction) over all players, reflecting that the goal of a university is to satisfy students as much as possible with their most preferred courses. Total market utility can be measured with a (traditional) cardinal utility objective function, an ordinal utility objective (comparing only the relative rankings of courses), or a binary utility objective (simply measuring the total number of seats assigned by the market).

Measures of Fairness

The range and standard deviation of individual student utility provide measures of dispersion among the students with high values indicating outcomes that would be viewed as highly unequal among students and, thus, unfair. As with allocative efficiency, fairness can be measured using cardinal, ordinal, or binary utility measures. For example, under binary utility, the range and standard deviation measure differences in the total number of courses awarded to a student. Significant dispersion in the number of courses assigned to a student is problematic given the basic expectation of getting a full or nearly full schedule of courses. Dispersion when using the other utility measures is similarly problematic.

Measures of Incentive Compatibility

To test the relative truth-inducing properties of the various course-allocation mechanisms, we adapt a common-value utility approach from Kominers et al. (2010), in which a student's true utility for each course is simulated as the mean of a private-value component and a common-value component. This setup creates a high degree of value correlation and a natural method to behave strategically under highly uncertain conditions: moving the weighting factor from the private signal toward the common-value signal tends to increase one's chances of getting highly popular courses. We implement these boundedly rational strategies using a range of weights (more and less extreme strategies) and among a higher and lower percentage of such "strategic" students. We can then measure the benefit or lack of benefit to honest and strategic students under a variety of scenarios, measuring when strategic play is most beneficial to those who employ such tactics and most hurtful to those who remain honest.

Focusing on these metrics, our results show a great deal of promise for mechanisms that use an integer-programming approach. Our five new algorithms represent new combinations of existing ideas from the literature, including round-by-round allocation as in draft mechanisms, top-trading cycles from matching theory, second-price concepts from auction theory, and the use of optimization both within rounds and globally across an entire market.

Round-by-round allocation has the potential to increase fairness by ensuring that no student gets allocated the student's $t+1$st course before all students have been allocated t courses (except when a student has exhausted the student's list of acceptable courses and, thus, can be allocated no more). Therefore, four of our five new algorithm variations investigate the use of round-by-round allocation for its potential to have good fairness properties.

Within the round-by-round framework, we pursue two different general approaches. The first emulates the top trading-cycle (TTC) algorithm commonly discussed alongside the deferred-acceptance algorithm in matching theory, adapted here to be iterated for the multiround allocation of several courses as opposed to matching theory's typical unit-demand setting. This algorithm is referred to as TTC for its connection to that classical approach.

Our second approach emulates auction theory in its use of second prices to squelch strategic manipulability. Although a Vickrey–Clarke–Groves (VCG) mechanism can be used to provide incentive compatibility in weakly dominant strategies even for the general combinatorial auction setting, it uses actual currency and a payment scheme to control incentives. For universities, however, course allocation cannot be settled through monetary bidding and payments (a situation known in the literature as nontransferable utility) making the control of incentives more difficult. Still, although a bidding student has no value for unused points after participation in the market, a multiround setting can potentially set up a situation in which unused bidding points in one round can be useful as points to be used in subsequent rounds, potentially returning some currency value and auction-like properties. We, therefore, investigate course allocation as a sequence of second-price auctions using bidding points in each round. This algorithm is referred to as SP for second price.

Each of these multiround algorithms is also implemented using within-round optimization, creating optimization versions of each: our third and fourth algorithm variations, TTC-O and SP-O, respectively. These within-round optimizations tend to improve efficiency in harmony with the multiround formats' focus on fairness, but the effect of this combined approach on incentives is not known, leading us to investigate the impact via simulation.

Our fifth algorithm attempts optimization over the entire market, first maximizing ordinal utility, followed by a second optimization to maximize cardinal utility subject to ordinal-utility optimality. This two-stage approach reflects a desire to avoid interpersonal cardinal-utility comparisons as much as possible, instead reflecting a university's desire to have a large number of highly ranked course assignments. The magnitudes of bid-point revelations are instead used only for tie breaking, guaranteeing that any cardinal-utility comparison is based on intrapersonal effects, not just direct interpersonal comparisons, which are harder to justify. Our findings show that this approach of ordinal-then-cardinal optimization (which we abbreviate OC) performs quite well in the areas of efficiency and fairness.

The remainder of the paper is organized as follow. First, a comprehensive review of the current body of knowledge is discussed in Section 2. Next, we introduce our five new algorithm variations for solving the course allocation problem in Section 3. Then in Sections 4 and 5, we discuss the results of two extensive simulations comparing these new algorithms to the most commonly discussed methods in the literature (the draft mechanism and bidding-point mechanism (BPM)) using the measures of efficiency, fairness, and incentive compatibility discussed. Concluding remarks are provided in Section 6.

Supporting material is provided in the online supplement. Online Appendix A describes the simulation details for the efficiency/fairness experiments in greater depth. Additional results on incentive compatibility are described in Online Appendix B, and a brief additional discussion of a more recent alternative mechanism (approximate competitive equilibrium for equal income (A-CEEI)) from Budish (2011) is given in Online Appendix C. In Online Appendix D, we explore robustness under a more random setup while varying the amount of correlation among student preferences, finding that the fairness of our round-by-round, optimization-based methods are more robust to extreme preference correlation as might be expected. All problem instances described in the paper as well as code for the generation of simulated instances are provided online at https://users.business.uconn.edu/bday/CAPfiles.html.

2. Background and Literature Review

A review of the literature reveals five prominent alternative algorithm variations for CAP: the bidding-point mechanism, the draft mechanism (including a "proxy bidding" version), random serial dictatorship, the Gale–Shapley Pareto-dominant market mechanism, and the A-CEEI mechanism. Each of these algorithms has its own strengths and weaknesses, and in the rest of this section, we mainly discuss the details of each. Moreover, at the end of this section, we also briefly review the other more recent papers for solving CAP.

Two main mechanisms have been most widely used by different universities, *BPM* and the *draft mechanism*. In BPM, a fixed bidding budget of points is assigned to each student and a bid is submitted that allocates these points among the student's various courses of interest. Following a complete submission from all students, all bids for all courses are sorted into a single list from highest to lowest and accepted, in turn, if eligible, one at a time. Each bid is considered if and only if the bidding student has not filled the student's schedule, the course does not conflict with the student's current assigned courses, and the course has not filled all its seats (Sönmez and Ünver 2010). This algorithm (with small variations) has been used by several schools, including University of Michigan's Ross School, Kellogg Graduate School of Management at Northwestern, Johnson Graduate School of Management at Cornell, Columbia Business School, Haas School of Business at University of California Berkeley, and Yale School of Management (Krishna and Ünver 2008).

Budish (2011) describes the main issue of BPM, its unfair results, in which one student may receive no course for one semester and the student's unspent points are wasted while another student may receive all of the student's desired courses, a result that unexpectedly happened frequently at University of Chicago's Booth School of Business. Indeed, BPM focuses only on maximizing allocative efficiency and does not consider equity among students. In fact, it compromises both equity in number of courses and fairness in value of courses among students to achieve a solution that greedily maximizes the total bid of assigned courses. Moreover, moving beyond a static, one-shot investigation for some highly popular courses, course prices may become distorted and chaotic if taken as signals for bidding from semester to semester. That is, the course price may become as high as the whole budget of a student in one semester, discouraging students in the next semester from bidding on it, only to see its price drop, leading demand to rise and fall sporadically over time.

In the draft mechanism, used by Harvard Business School beginning in the mid-1990s, a computer takes students' preferences over individual courses. Then student proxies take turns in a "draft order" with one available course seat assigned to each student in each round. In the draft-order procedure, students are first randomly ordered, and in even-numbered rounds, their order is reversed (Budish and Cantillon 2012).

The draft mechanism is fairer than BPM because the round-by-round allocation naturally decreases the range of the number of assigned courses per student. However, the initial random favoritism among students may result in successive bad luck for a student and causes dissatisfaction based on inequality of ordinal or cardinal utility. In fact, the draft mechanism only collects the student ordinal preferences and, therefore, cannot distinguish between a slight difference and a significant difference among preferences. Hence, a student who is almost indifferent between two courses C1 and C2 and has put C1 first may get this course while another student who wants C1 much more than C2 may lose it. Although a small decrease in utility of the first student could allow a large increase for the second, this opportunity to improve overall cardinal utility is lost by the draft mechanism.

Furthermore, in both BPM and the draft mechanism, each student's belief about others' favorite courses affects the student's declared preferences, encouraging the student to behave strategically and misreport the student's preferences (Budish and Cantillon 2012). In fact, our simulations show that BPM is seriously vulnerable to student strategic behavior with an intentional misreporting of preferences easily leading to a more favorable outcome.

Kominers et al. (2010) introduce a variation of the draft mechanism based on proxy bidding, which showed improved performance with respect to incentives in their simulations, that is, showing less benefit to those who choose to bid strategically. The approach is to generate true-bid values for simulated students based on an equally weighted average of a common value and a personal value for each course and moving the weight toward the common (popularity-based) value to simulate strategic behavior. We adapt this approach to the current setting in the experiments described in Section 4.3 as a first approximation of strategic behavior in a setting of bounded rationality (incomplete and uncertain information with no ability to devise a stochastically optimal strategy) as in their paper. Although their results show some improvement of incentives over the draft mechanism, it suffers some of the same problems, in which a commitment to a random prioritization of students results in lost opportunities for efficiency improvements as mentioned. Also, some of their results are based on the assumption of nonoverlapping courses although our current study takes the more realistic assumption that some course sections overlap in time and that different sections of the same course may be available although only one can be taken.

A third algorithm discussed in the literature is the random serial dictatorship, in which students are sorted randomly, and in each turn, one student picks the student's entire bundle of courses from any seats still available. This mechanism is the only incentive-compatible mechanism for CAP in the literature (see Papai (2001), Ehlers and Klause (2003), and Hatfield (2009)) because student strategic behavior cannot change the assigned courses. However, fairness and equity are totally neglected in this method (Budish and Cantillon 2012). Indeed, randomness of this sort is in favor of incentive compatibility in general: basing "prioritization" on declared preferences (rather than randomness) in general introduces opportunities for manipulation of the declared preferences. But, given the quite bad fairness and efficiency for these totally random prioritizations, the perspective of the current paper is to find mechanisms that do not perform very badly on incentives while offering drastic improvements on fairness and efficiency.

A fourth algorithm is the Gale–Shapley (GS) Pareto-dominant market mechanism, which was developed by Sönmez and Ünver (2010) as a hybrid between BPM and the familiar GS deferred-acceptance algorithm for matching. This algorithm gives a fixed bidding budget to each student. Then, each student submits both a rank-ordered list and a bid list for the student's desired courses, and these two lists need not agree with each other. (If the rank-ordered lists and the bid lists agree with each other, GS and BPM result in the same solution.) In this version of the GS algorithm, first each student (proxy) points to the student's most preferred course based on rank and then proposes a match. Then, each course sorts all the offers and keeps the highest bids, up to capacity, and rejects the remaining ones. Students with rejected offers point to their next most preferred course, proceeding as in a deferred-acceptance algorithm with courses keeping the current set of highest bids and rejecting those that don't fit. This algorithm stops when no student is rejected, at which point each student can take the courses that hold the student's name.

Krishna and Ünver (2008) show that, based on the theory and the results of their field experiments, asking for a rank-ordered list as well as a bid list increases the efficiency of results of the GS algorithm in comparison with BPM. However, getting two sets of information that may not agree with each other may, in reality, increase the risk of misreporting preferences and encouraging students to behave strategically. Furthermore, the GS method, as with other bidding mechanisms, at times results in solutions assigning no course to some students.

A fifth algorithm for CAP was introduced by Budish (2011), the A-CEEI mechanism. In A-CEEI, students report their ordinal preferences over all possible schedules of courses, and they have the option to report either their additive or nonadditive complementary/substitutability preferences, depending on the specific implementation. Then, it assigns a random, approximately equal budget to students and allocates courses through a series of optimizations based on student preferences, student budgets, and course prices, emulating a Walrasian-style price equilibrium. Although several algorithmic variations are possible, one concrete implementation uses a metaheuristic tabu search method to find a nearly optimal price set, which approximately equates the number of available seats of each course (supply) with its demand. The course demand for each price set is calculated by allocating each course seat to a student who wants it and can afford it. The final allocation of courses is achieved when the error, the difference between allocated and available seats, is less than a predetermined tolerance. Accordingly, A-CEEI allows infeasible solutions in which the number of assigned students may exceed the course capacity, a problem to be sorted out after the mechanism. Also, similar to the draft mechanism, this method randomly prioritizes students with randomly generated budget inequality directly affecting the results, which should, in principle, be avoided whenever possible.

Furthermore, finding an A-CEEI price-set that approximately balances the demand and supply of the market for real-size problems can be a very time-consuming computational process. Indeed, because we were unable to create a fast implementation of the A-CEEI approach subject to nonoverlap constraints, we were unable to conduct a direct comparison of A-CEEI to our own mechanisms, which, in contrast, were easily implemented hundreds of times in our simulations. Further discussion of A-CEEI relative to the current context is provided in Online Appendix C.

Along with these main algorithms, recently two other streams of research have been presented to solve CAP. In the first stream, another definition of CAP has been studied. Diebold et al. (2014) defined CAP differently and assumed that each course or organizer also has preferences over students. Based on their definition, CAP is a two-sided matching problem (Roth and Sotomayor 1990). Therefore, they defined a stable matching of courses and students and compared the first-come, first-served procedure and the GS mechanism with their mechanism. However, they did not consider overlapping constraints in their model and solutions. Nogareda and Camacho (2016) also had the same approach and defined CAP as a two-sided matching problem, and they did not consider overlapping courses either.

In the second stream, first Budish et al. (2013) proposed a random allocation mechanism to solve general matching problems, including CAP. However, their mechanism cannot consider the section and timeslot overlapping constraints simultaneously. Also, design of their mechanism for a multiunit market is similar to Budish (2011), and both need to find a set of prices balancing supply and demand and clear the market. Finding this set of prices as we discussed for Budish (2011) is computationally very difficult. Recently, Akbarpour and Nikzad (2015) and Nguyen et al. (2014) proposed other random allocation mechanisms that, similar to Budish (2011) and Budish et al. (2013), terminate with infeasible solutions. Here, our new methods all compute feasible solutions.

3. New Algorithms and Metrics for CAP

The course allocation problem (CAP) consists of a set of courses, some of which may have more than one section/time offered. Define the set $C = \{1,..,m\}$ as set of all course sections offered in the market in which each course section j has q_j seats. The course-section seats are allocated to a set of students $I = \{1,..,n\}$, each of whom can take at most k courses in each semester.

By standard assumptions of rationality, each student i has real cardinal preferences (utility) over course sections u_{ij}. In any of our cardinal-eliciting mechanisms, the student submits the student's cardinal preferences (i.e., bids) $b_{ij} \geq 0$ for each course section with $\sum_j b_{ij} \leq 1{,}000$. (As is standard in utility theory, affine transformations do not affect preferences, so it is always possible to normalize an approximately truthful revelation to this 1,000-point bidding scale. Higher budgets would allow higher precision.)

In some mechanisms, ordinal preferences are submitted; each student i submits the student's integer ordinal preferences over course sections r_{ij}, where higher r_{ij} shows a more preferred course section. (Although lower is better seems more common in plain language, with $r_{ij} = 1$ indicating the first-best course, etc., this approach is not practical for optimization because nonallocation would seem lower and, therefore, better.) For our computations, we let $0 \leq r_{ij} \leq 100$ although other upper bounds provide similar results. Unlike the GS algorithm for CAP in Sönmez and Ünver (2010), we require consistency among any student's course ranks and bids, that is, $\forall j, j', r_{ij} \leq r_{ij'} \Leftrightarrow b_{ij} \leq b_{ij'}$. In practice, for the new algorithms, which use both cardinal and ordinal measures, a student need only submit the student's cardinal preferences, and then the student's ordinal preferences may be inferred because of this consistency requirement.

Each student will get a feasible bundle of course sections S_i in which no two course sections overlap, none are different sections of the same course, and the number of course sections in the bundle is less than or equal to k. We define the binary decision variable x_{ij} to one if and only if a seat in course section j is assigned to student i.

To define forbidden course overlap, let binary parameter $O_{jj'}$ equal one if and only if the course section j and j' are overlapping course sections in time or they are different sections of a same course. Therefore, a feasible bundle of course sections, S_i, is defined by the corresponding x_{ij}, which must satisfy $\sum_{j=1}^{m} x_{ij} \leq k$ and $x_{ij} + x_{ij'} \leq 1$ for all (j, j') with $O_{jj'} = 1$. Although it is possible to require transitivity, in which $(O_{jj'} = 1)$ and $(O_{j'j''} = 1)$ would imply $(O_{jj''} = 1)$, we do not make such an assumption. (Computational experiments that included a transitivity assumption were easier to solve, but we abandoned this line of inquiry given that overlap based on the preponderance of nontransitivity in practice; one course might overlap with another in time but not an alternative section of the same course, for example.)

The utility of the set S_i is defined as the sum of the utility of its members; that is, we assume additive utility over course sections. Accordingly, the "best bundle of courses" for student i or S_i^* is also defined as a feasible bundle of course sections for student i that maximizes the student's total utility.

For round-by-round algorithms, the letter t is added as a round index subscript, giving us q_{jt} for the available seats in course j in round t and x_{ijt} for an allocation of a seat in course j to student i in round t. The set of eligible course sections for student i in round t is denoted E_{it}. For round 1, we initialize $E_{i1} = \{j : b_{ij} > 0\}$ with $E_{i,t+1}$ formed by successively removing courses no longer in play at the end of each round. Also, p_{jt} is defined as the price of course j in round t, which is needed for one of the proposed algorithms.

For simplicity, in the rest of this paper, the term "course" refers to the "course section" unless it is explicitly mentioned.

Turning our attention from notation to the specific metrics of interest, market efficiency is measured by the following integer programming formulation:

$$\text{Max} \sum_{i,j} u_{ij} x_{ij}, \tag{1}$$

$$\sum_{i} x_{ij} \le q_j \qquad \forall j, \tag{2}$$

$$\sum_{j} x_{ij} \le k \qquad \forall i, \tag{3}$$

$$x_{ij} + x_{ij'} \le 1 \quad \forall j, j' \text{ with } O_{jj'} = 1, \tag{4}$$

$$x_{ij} \in \{0, 1\} \qquad \forall i, j. \tag{5}$$

The objective is to maximize total (cardinal) utility for the market with no more seats awarded than are available in any course (2), no more than k courses per student (3), and no student taking conflicting courses (4). Any allocation mechanism satisfying (2)–(5) can be compared using the following natural efficiency and fairness metrics as discussed in Section 1 and summarized in Table 1. Typically, in practice, market efficiency based on true utility u_{ij} is not possible because only bid values b_{ij} are given. So, for any of these metrics, both revealed and actual metrics are possible based on the situation, the latter typically only possible via simulation or controlled experiment.

3.1. Multiround TTC Algorithm for CAP

The top trading-cycle algorithm was developed to solve the house-allocation problem, which finds a new assignment of houses among a set of house owners who are selling their houses, and each owner has preferences over other houses in the market. In this algorithm, each owner points to the owner's most preferred house, and each house points to the highest offer. If there is a cycle (i.e., a house points to an owner, and the owner also points back to the same house), the owner is assigned to the house, and both are removed from the market. The process then reiterates with the remaining participants. Another version of this algorithm assigns students to schools and has been used in practice (see Schummer and Vohra 2007 and Morrill 2012). In the current paper, we modify this algorithm to solve the course-allocation problem.

Our new TTC algorithm for CAP attempts to overcome the weaknesses of BPM and the draft mechanism while keeping their strengths. BPM considers the cardinal bids of students and avoids random favoritism of students when it is not needed; however, it may assign all top preferred courses to some students and assign no courses to others. On the other hand, the draft mechanism randomly prioritizes students and assigns courses to students based on the random order. Nevertheless, it considers fairness as no student can get that student's $t + 1^{st}$ course before all other students get their t^{th} course. TTC is developed to combine these strong points. Accordingly, it assigns one course in each round to each student, but the winning students are defined based on the students' bid values and not based on random numbers.

The details of this algorithm are as follows:

(1) Each student is given a fixed budget (e.g., 1,000 points) and submits bid points b_{ij} for each course with the sum not exceeding the budget.

(2) Round t, the algorithm acts on behalf of each student as follows:

(i) Each student points to $j_i^* = \text{argmax}_{j \in E_{it}} b_{ij}$ and offers the bid amount ($b_{ij_i^*}$) for it.

(ii) Each course accepts up to q_{jt} of the highest offers and rejects any remaining offers. Then capacities are updated: $q_{jt+1} = q_{jt} - \sum_{i=1}^{N} x_{ijt}$.

(iii) $E_{i,t+1}$ is calculated based on the following recursive equation:

$$E_{i,t+1} = E_{it} - \{j_i^*\} - \{j : x_{ij_i^*t} = 1, O_{jj_i^*} = 1\} - \{j : q_{jt+1} = 0\}.$$

(iv) If there are rejected students, they will repeat steps (i)–(iii) until no more students are rejected. Accordingly, all students take *one* more course in round t or leave the market because there are no courses in their E_{it}.

(3) Step 2 repeats until all students take k courses or leave the market.

Example 1. Consider a set of students $I = \{S1, S2, S3, S4\}$ and courses $C = \{C1, C2, C3, C4, C5\}$ with $\{2, 3, 3, 2, 2\}$ seats, respectively, where C1 and C4 are overlapping courses and each student can take at most three courses. Table 2 shows the submitted student bid lists

Table 1. Efficiency and Fairness Metrics

Category	Efficiency Larger ⇒ better	Fairness Smaller ⇒ better	
Cardinal	$\sum_{i,j} u_{ij} x_{ij}$	$\text{Range}_i(\sum_j u_{ij} x_{ij})$	$\text{StDev}_i(\sum_j u_{ij} x_{ij})$
Ordinal	$\sum_{i,j} r_{ij} x_{ij}$	$\text{Range}_i(\sum_j r_{ij} x_{ij})$	$\text{StDev}_i(\sum_j r_{ij} x_{ij})$
Binary	$\sum_{i,j} x_{ij}$	$\text{Range}_i(\sum_j x_{ij})$	$\text{StDev}_i(\sum_j x_{ij})$

Table 2. Student Rankings and Bid Lists for Example 1

Student 1		Student 2		Student 3		Student 4	
Rank	Bid list	Rank	Bid list	Rank	Bid list	Rank	Bid list
C1	400	C3	256	C4	245	C1	251
C3	230	C2	252	C1	243	C3	242
C4	200	C4	246	C3	240	C2	235
C2	150	C1	245	C2	230	C4	201
C5	20	C5	1	C5	42	C5	71

(cardinal preferences) and rank-ordered lists (ordinal preferences) for the offered courses. In Example 1, we assume that $1 \leq r_{ij} \leq 5$ and higher r_{ij} represents more preferred courses.

Under TTC, students first point to their top-ranked course. Therefore, C1 receives two offers from S1 and S4, and based on its capacity accepts theses offers. At the same time, C3 and C4 receive one offer from S2 and S3, respectively, and they accept these offers. Thus, round 1 ends with one course assigned to each student and C1 with no more seats, leaving the market. The updated course capacity list is $\{0, 3, 2, 1, 2\}$.

In round 2, C2 accepts one offer from S2, but C3 receives three offers from S1, S3, and S4, which is more than its capacity. Then C3 fills its two available seats from the two highest offers (242 and 240) and rejects the lowest one (200). Therefore, S1's offer for C3 is rejected, but S1 gets another chance and S1's next course in S1's rank-ordered list. S1 points to the fourth course in S1's rank-ordered list, skipping C4 because of overlap with C1 already in S1's schedule, and S1's offer for C2 is accepted. Each student has two courses, so round 2 ends with the updated course capacity list $\{0, 1, 0, 1, 2\}$.

In round 3, S1 and S2 point to their next highest ranked courses among available courses, C5 and C4, respectively, and because there are enough seats, their offers are accepted. At the same time, C2 receives two offers from S3 and S4, but with only one available seat, it only accepts the higher bid offer (235) and rejects the lower one (230). Accordingly, S3, who is rejected by C2, gets another chance and points to S3's next highest ranked course with available seats, C5, and because it has an available seat, S3 takes the course. With no course seats available, the final course assignment for Example 1 and the total rank of assigned courses to each student is shown in Table 3. Also, Table 4 shows the efficiency and fairness metrics for Example 1 solved by TTC.

3.2. SP Algorithm for CAP

As mentioned, the VCG mechanism can elicit truthful bids from bidders in an auction, in which monetary bids are used to compute winners and payments for goods received. However, in CAP, monetary transfers cannot be used, making it more difficult to control incentives; in bidding with points (a fictional currency used only for the CAP process), receiving low prices or keeping unused points has no value to a bidding student, and therefore, incentives are drastically different.

Table 3. Final Solution to Example 1 Solved by TTC

Assigned courses	Student 1	Student 2	Student 3	Student 4
Round 1	C1	C3	C4	C1
Round 2	C2	C2	C3	C3
Round 3	C5	C4	C5	C2
Total cardinal utility	570	754	527	728
Total ordinal utility	8	12	9	12
Total binary utility	3	3	3	3

Table 4. Efficiency and Fairness Metrics of Final Solution to Example 1 Solved by TTC

		Fairness	
	Efficiency	Range	Standard deviation
Cardinal	2,579	227	97.88
Ordinal	41	4	1.79
Binary	12	0	0

But, in a multiround setup, we explore the possibility that returned points may retain some currency value from round to round, encouraging truthful bidding. For example, suppose a bidder would like to bid aggressively for the student's top course, say 350 points, only to find later that the student could have bid 250 and still would have been awarded the course. Retrospectively, the student would wish to have bid lower for this course so that more of the bid points could have been used on subsequent course bids. This suggests an incentive for the student to strategically lower the student's truly high value for the course. But, if a round-by-round mechanism returned points that exceeded the threshold necessary to "buy" the course with points (i.e., gave her back the 100-point excess), the student would not have such an incentive.

Here, we explore this notion of returning points, similar to discounts in a second-price auction, to be used across rounds. In a setting with unit-demand bidders, the VCG mechanism becomes a uniform-price auction in which the top q bidders win the q available items and pay the amount of the $q+1^{st}$ bid, which we emulate in each round of our multiround setup. The resulting algorithm is called the SP algorithm for CAP. The algorithm is very similar to TTC, but with step 2, part (ii) replaced as follows with (ii)^{SP}:

(ii)^{SP}. Each course accepts up to q_{jt} of the highest offers and rejects any remaining offers. Then capacities are updated: $q_{jt+1} = q_{jt} - \sum_{i=1}^{N} x_{ijt}$.

If $q_{jt+1} = 0$, then let price p_{jt} equal the highest rejected offer for j or else let $p_{jt} = 0$. Return unspent points to use on the next most preferred course, that is, let $j_i^{next} = \mathrm{argmax}_{j \in E_{it} - \{j_i^*\}} b_{ij}$ and let $b_{ij_i^{next}} = b_{ij_i^{next}} + b_{ij_i^*} - p_{jt}$.

We next consider the same market as discussed in Example 1 but now under SP. As before, C1 receives two offers, C3 receives one offer, and C4 receives one offer in the first round. Because there are no rejected offers, the prices of all courses in this round are set to zero, and the points of all offers are added to the next course in their rank-ordered list. In round 2, courses begin to reach capacities and, thus, charge prices for courses, only returning the unused portion of each bid, which is then used for the next set of offers. These next offers will occur before round 2 ends because some students were rejected from their second favorite course and must be awarded a course before the next round begins.

We omit the full details, but the results of SP on Example 1 with efficiency and fairness metrics are given in Tables 5 and 6, respectively. Note that relative to the TTC results in Table 3, this final allocation awards the same number of courses and has the same ordinal efficiency (same total rank). However, other measures tell different stories: the ordinal fairness metrics (both range and standard deviation of rank) are lower (better) for SP, and the cardinal fairness metrics (range and standard deviation of cardinal utility) are both lower (better) for TTC. Also, total cardinal utility is larger for SP, indicating overall that it is difficult to rank allocations/algorithms based on the many design metrics even for a very small (oversimplified) problem.

3.3. The Optimized Multiround Algorithms: TTC-O and SP-O

The TTC and SP algorithms locally maximize the benefit of each student in each round in a greedy fashion, not considering that a better outcome may be available for the whole market even within the same round. Instead, at times exchanging a student's r^{th} ranked course with the student's $(r+1)^{st}$ course may benefit two or more students. In this case, the total rank for the whole market can be improved with two or more made better off and one student worse off. Considering the benefit to the entire market in each round is made possible in the following optimization versions of the TTC and the SP mechanisms, which we call TTC-O and SP-O.

In both methods (and as in our fifth and final method, OC), we make use of a two-stage optimization, first finding the maximum possible total rank and then maximizing total bid points among all rank-optimal solutions because there are typically many. The optimizations for TTC-O and SP-O occur in each round in contrast to OC.

Table 5. Final Solution to Solved by SP

Assigned courses	Student 1	Student 2	Student 3	Student 4
Round 1	C1	C3	C4	C1
Round 2	C3	C2	C3	C2
Round 3	C2	C4	C5	C5
Total cardinal utility	780	754	527	557
Total ordinal utility	11	12	9	9
Total binary utility	3	3	3	3

Table 6. Efficiency and Fairness Metrics of Final Solution to Example 1 Solved by SP

		Fairness	
	Efficiency	Range	Standard deviation
Cardinal	2,618	253	113.37
Ordinal	41	3	1.30
Binary	12	0	0

Multiround TTC-O. In round t, we solve optimization models (6)–(9) followed by (10) and (11):

$$Z_{t1} = \text{Max} \sum_{j,i \in E_{it}} r_{ij}\, x_{ijt}, \tag{6}$$

$$\sum_i x_{ijt} \le q_{jt} \quad \forall j, \tag{7}$$

$$\sum_{j \in E_{it}} x_{ijt} \le 1 \quad \forall i, \tag{8}$$

$$x_{ijt} \in \{0,1\}, x_{ijt} \in E_{it}, \tag{9}$$

$$Z_{t2} = \text{Max} \sum_{j,i \in E_{it}} b_{ij}\, x_{ijt}, \tag{10}$$

$$\sum_{j,i \in E_{it}} r_{ij}\, x_{ijt} = Z_{t1}. \tag{11}$$

Notice that (6)–(9) represent an instance of the transportation problem, known to have a totally unimodular constraint matrix, thus making it easy to solve as a relaxed integer program. The addition of constraint (11) to lock in the total rank at its maximum value and changing the objective to (10) in the secondary optimization do not make the problem harder in practice.

Consider again the market discussed in Example 1. In the first round of TTC-O, just as in TTC, courses C1, C3, C4, C1 are assigned to S1, S2, S3, S4, respectively. (Because all can get their top-ranked course, it is optimal to do so in the first round.) But, in round 2, the TTC-O method differs, with C3, C2, C3, C2 assigned to S1, S2, S3, S4, respectively. In TTC, S4's bid for C3 beats out S1's offer for this course, but TTC-O recognizes that denying the seat to S4 results in S4 getting a course of only one lower rank, whereas denying the seat to S1 would push S1 two steps down S1's rank list, a worse move overall for total rank in this round. (The third round also proceeds differently although details are omitted.) The final assignment of courses to students with the TTC-O algorithm and efficiency and fairness metrics are presented in Tables 7 and 8, respectively.

Comparing Table 7 to the results of TTC and SP (Tables 3 and 5), we see a third distinct allocation of courses to students delivering the same ordinal efficiency of 41. The results in Table 7 have a higher (better) cardinal utility; also, we see that its allocation performs at least as well if not better in every other fairness and efficiency metric.

Table 7. Final Solution to Example 1 Solved by TTC-O or SP-O

Assigned courses	Student 1	Student 2	Student 3	Student 4
Round 1	C1	C3	C4	C1
Round 2	C3	C2	C3	C2
Round 3	C5	C4	C2	C5
Total cardinal utility	650	754	715	557
Total ordinal utility	10	12	10	9
Total binary utility	3	3	3	3

An SP-O. In the SP-O algorithm, at the end of each round, the excess points are returned to students and applied to their next-round offers. To calculate the extra points, we need to determine the price of each course in each round. The dual optimization problem (12)–(14) followed by (15) and (16) at the end of round t helps us to find the course prices in this round.

$$W_{t1} = \text{Min}\left(Z_{t1}D + \sum_j q_{jt}\, p_{jt} + \sum_i v_{it}\right), \tag{12}$$

$$p_{jt} + v_{it} + r_{ij}\, D \geq b_{ij}, \tag{13}$$

$$p_{jt}, v_{it} \geq 0, D \text{ is unrestricted}, \tag{14}$$

$$W_{t2} = \text{Min} \sum_j q_{jt}\, p_{jt}, \tag{15}$$

$$Z_{t1}D + \sum_j q_{jt}\, p_{jt} + \sum_i v_{it} = W_{t1}. \tag{16}$$

In these models, p_{jt} is the price of course j in round t, and v_{it} can be described as the value of student i in round t in the market. Also, in (15), we find a set of prices that minimize the weighted summation of course prices. It means that we are interested in decreasing the price of courses with fewer seats less than courses with more seats. This objective function, along with (16), which locks (12) at its optimum value, results in finding unique optimum values for the course prices in each round.

Consider again the market discussed in Example 1. Because this is a very small example, returning points to students does not change the final solutions, and the assigned courses in each round of TTC-O and SP-O and their final solutions are the same. Therefore, the final assignment of courses to students with the SP-O algorithm and efficiency and fairness metrics are presented in Tables 7 and 8, respectively.

Table 8. Efficiency and Fairness Metrics of Final Solution to Example 1 Solved by TTC-O or SP-O

		Fairness	
	Efficiency	Range	Standard deviation
Cardinal	2,676	197	74.58
Ordinal	41	3	1.09
Binary	12	0	0

3.4. OC Algorithm for CAP

In the OC algorithm for CAP, we maximize ordinal utility followed by maximizing cardinal utility among rank-maximal solutions as in TTC-O and SP-O but here performing this two-part optimization once for the whole market instead of in each round. To define these models formally, we use the general market feasibility constraints (2)–(5).

$$Z_1 = \text{Max} \sum_{i,j} r_{ij}\, x_{ij}, \tag{17}$$

$$(2)\text{–}(5)$$

$$Z_2 = \text{Max} \sum_{i,j} b_{ij}\, x_{ij}, \tag{18}$$

$$\sum_{i,j} r_{ij}\, x_{ij} = Z_1. \tag{19}$$

$$(2)\text{–}(5)$$

Tables 9 and 10 display the final assignment of courses to students as well as efficiency and fairness metrics under OC using a maximum schedule size of k = 3, respectively. However, if k changes and increases to four, the solution of OC will change although the solution of other presented algorithms stay the same. Tables 11 and 12 show the final assignment as well as the efficiency and fairness metrics of courses when k = 4, respectively. Both solutions achieve a total rank of 42, one unit higher than any previously discussed allocation, with the former showing that it is possible to achieve this better solution while maintaining equal schedule sizes (binary fairness metrics equal to zero). Although this small change shows a potential issue of fairness in OC results, our simulation results show that the OC algorithm is able to achieve the lowest binary fairness metric among all algorithms.

3.5. Pareto (In)Efficiency

Given these five new algorithms, it is natural to ask whether each satisfies Pareto efficiency with respect to submitted preferences. It is easy to see that optimization over the entire market as in (1)–(5) results in Pareto efficiency with respect to the objective function used: cardinal, ordinal, or binary. (This follows because the existence of a feasible Pareto improvement would contradict optimality.) Thus, OC guarantees ordinal Pareto efficiency because of its first optimization.

Table 9. Final Solution to Example 1 Solved by OC Using $k = 3$

	Student 1	Student 2	Student 3	Student 4
Assigned courses	C1, C3, C5	C3, C2, C4	C4, C2, C5	C1, C3, C2
Total cardinal utility	650	754	517	728
Total ordinal utility	10	12	8	12
Total binary utility	3	3	3	3

Table 10. Efficiency and Fairness Metrics of Final Solution to Example 1 Solved by OC Using $k = 3$

		Fairness	
	Efficiency	Range	Standard deviation
Cardinal	2,649	237	92.18
Ordinal	42	4	1.66
Binary	12	0	0

Although an OC solution may not be cardinal Pareto efficient, any Pareto-improving solution in cardinal utility must necessarily degrade the total ordinal utility of the market (harming at least one student in ordinal preferences) by virtue of its ordinal-then-cardinal optimization. Thus, solutions that improve the situation on both metrics are not available. Sadly, the same guarantees cannot be made for the other mechanisms as can be shown via counter Example 2 described in Table 13.

Example 2. Consider a market with two students S1 and S2, each of whom can take at most two courses ($k = 2$) and a set of five courses $C = \{C1, C2, \ldots, C5\}$, each with one seat. Let C1 and C3 be overlapping courses, and let the preferences of S1 and S2 be as displayed in Table 13.

Under each of the mechanisms discussed here, with the exception of OC, a Pareto-inefficient solution is selected, considering either ordinal or cardinal utility. Under the draft mechanism, BPM, TTC, or SP, student 1 gets {C1, C5}, and student 2 gets {C2, C4}. Alternatively, under TTC-O and SP-O, students 1 and 2 get {C1, C4} and {C2, C3}, respectively. But OC, on the other hand, awards them {C2, C3} and {C1, C4}, respectively, which is a (cardinal or ordinal) Pareto improvement over either outcome. Note also that straightforward cardinal-utility maximization results in an alternate cardinal Pareto efficient solution, {C3, C4} and {C1, C2}, but OC tends to reject this more lopsided solution, which seems to place too much weight on small cardinal differences as compared with the relative equity implied in ordinal comparisons. Note also that this latter cardinal Pareto-efficient solution fails to be ordinal Pareto efficient, and the OC solution is Pareto efficient in both senses.

Table 11. Final Solution to Example 1 Solved by OC Using $k = 4$

	Assigned courses			
	Student 1: C1, C3	Student 2: C3, C2, C4	Student 3: C4, C2, C5	Student 4: C1, C3, C2, C5
Total cardinal utility	630	754	517	799
Total ordinal utility	9	12	8	13
Total binary utility	2	3	3	4

Table 12. Efficiency and Fairness Metrics of Final Solution to Example 1 Solved by OC Using $k = 4$

		Fairness	
	Efficiency	Range	Standard deviation
Cardinal	2,700	282	110.23
Ordinal	42	5	2.06
Binary	12	2	0.71

4. Efficiency and Fairness Results

To assess the proposed algorithms and compare their results with BPM and the draft mechanism, we randomly generated 100 sample markets. Each consists of $n = 900$ students, each of whom can take up to $k = 6$ courses, and there are 83 courses, some with multiple sections, for a total of $m = 112$ course sections. (We assumed two courses having five sections, one course with four sections, two courses with three sections, 14 courses with two sections, and 64 courses with one section.) These features were selected to roughly approximate the characteristics of offered courses at Harvard Business School,[1] a prominent user of algorithmic course allocation. Accordingly, the capacity of 10%, 30%, and 60% of course sections were drawn from discrete uniform distributions with probabilities of {15, 25, 35}, {40, 50, 60}, and {70, 80, 90, 100}, respectively. To establish overlap among course sections, eight weekly time slots were generated, and course sections were considered overlapping if they shared a time slot or were sections of the same course. Also, similar to HBS, we generated 10 different major fields of study to group correlated courses and based student preferences on these fields of study. Further details on how these preferences were generated are given in Online Appendix A. All experiments in this paper involving optimization used CPLEX version 12.1 on a 2.53-GHz machine.

In this first set of experiments, we assumed that the generated cardinal and ordinal preferences are true student preferences, and they have not been manipulated to improve one's allocation. (In Section 5, we relax this truthfulness assumption in a second set of experiments.) Here, we present and interpret values for the efficiency and fairness metrics proposed in Table 1.

4.1. Binary Efficiency and Fairness

Recall that our notion of binary utility measured a one for an assigned seat and zero otherwise. Averaging total binary utility of the market over each of 100 simulated markets, therefore, shows us the

Table 13. Bids Showing Pareto Inefficiency Under Course Overlap

	Bid lists	
Courses	Student 1	Student 2
C1	385	380
C2	320	350
C3	180	100
C4	105	120
C5	10	50

average number of assigned seats by each mechanism. With 900 students seeking six courses in each case, each market has 5,400 opportunities to assign students to a course. For easier interpretation, Figure 1 shows the average number of missed opportunities (amount below 5,400) for each algorithm. Clearly, the optimization-based algorithms do a better job of getting students into seats with OC doing a very good job, missing only six opportunities out of 540,000 opportunities to place students.

Turning next to the range of binary utility across students as a metric of fairness, we note that, in almost every random trial, at least one student received a full six-course schedule. Thus, the range is most indicative of how the worst student is treated in terms of the number of courses the student is assigned. Figure 2 shows the average of the interstudent range across simulation instances.

As Figure 2 depicts, in BPM, the range is 2.05, on average; that is, the student with the lightest schedule gets only 3.95 courses on average. Although BPM assigns three or fewer courses to at least one student for 12 out of 100 instances, this occurred for only one instance under TTC or SP. But, again under this metric, the optimization-based approaches do best; none ever assigned fewer than four courses at worst, and OC never assigned fewer than five courses to students in any instance. Indeed, in 98 of 100 instances, OC assigned six courses to every single student, and BPM did not do so (complete 5,400 assignments) in even one instance. Results for the standard deviation of binary utility were qualitatively quite similar to those presented in Figure 2 and are, thus, omitted.

4.2. Ordinal Efficiency and Fairness

Figure 3 shows the simulation results of the different mechanisms based on the average total rank (ordinal efficiency) of assigned courses. As BPM scored the worst on this metric, the figure shows amounts above this worst benchmark case. Clearly again, the optimization-based methods dominate the other algorithms with OC in a strong lead, getting much better outcomes in terms of how students order the courses.

Equity in the rank of assigned courses gives us our next measure of fairness across students, as shown in Figure 4. Because, in almost all instances, the maximum total rank of assigned courses to a student is equal to the maximum possible 585 for all algorithms considered, the range of ordinal utility is indicative of the total rank of the worst student and how the student's total rank compares to this maximum. Note that the total rank equal to 585 means that a student could take the student's (100, 99, 98, 97, 96, 95) ranked courses. In other words, the student could take the six highest ranked courses on the student's ordered list. Moreover, the worst total rank possible for a student with a full schedule in these simulations (given the artificial 100-point ordinal basis and bidding on a maximum of 35 different courses) is 411, corresponding to the student's ordinal objective value when the student receives the student's 30th, 31st, ... 35th best courses. Thus, ranges greater than $585 - 411 = 174$ represent a worst student necessarily getting a reduced course load. Also note that this measure does grow quickly: a worst student getting the student's 2nd, 3rd, ... 7th courses (the student's second best possible schedule, the second best possible market measure) indicates a range of six, and simply removing one course from the worst-off student increases the range by at least 66 (the ordinal objective value of a 35th best course).

Figure 4 again shows the same general pattern with BPM scoring worst, optimization-based algorithms

Figure 1. (Color online) The Average Number of Missed Assignment Opportunities per Algorithm

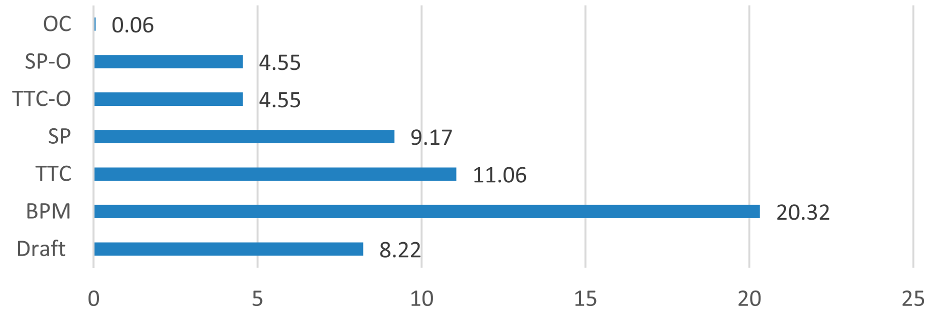
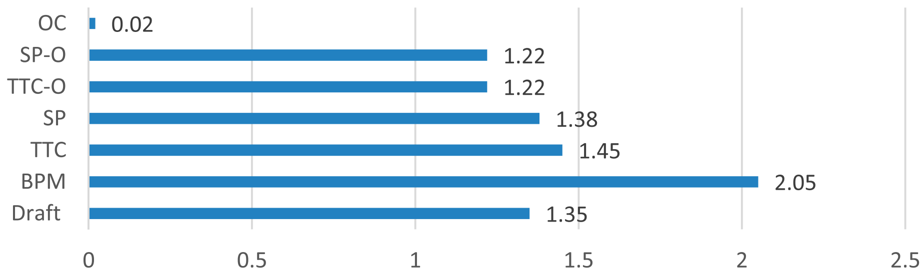


Figure 2. (Color online) The Average Range of Binary Utility per Student per Algorithm



doing best, and OC in a clear lead compared with all others. Looking at Figure 5, which shows average ordinal fairness performance via the standard deviation, we see a similar pattern but with less contrast, on average, across algorithms than the extreme cases measured by the range.

4.3. Cardinal Efficiency and Fairness

Figure 6 displays the average cardinal utility from each student's assigned courses under each mechanism. As Figure 6 depicts, the draft mechanism achieves the lowest average cardinal utility with OC and BPM achieving the first and second best on this metric, respectively. Although BPM does well on this metric, comparing Figure 6 to previous metrics, we see that BPM gets a high cardinal utility by grabbing at it greedily, sacrificing fairness and efficiency among the previous metrics. In this context, OC proves that this sacrifice is unjustified; it is possible (as OC does) to score better than BPM in all categories.

Indeed, Figure 7 displays the average range of cardinal utility, and we see that BPM gets a high overall cardinal utility by increasing the inequality between the top and the bottom of the market. Again we see optimization-based algorithms dominating the others, but here SP-O and TTC-O edge out OC in contrast to the previous results we have seen. OC seems willing to increase the range of cardinal preferences to keep the ordinal measures more equitable, consistent with its ordinal-then-cardinal formulation.

Figure 8 again shows the standard deviation, painting a similar picture to that of the range, again favoring the optimization-based algorithms and relaying bad news for BPM.

5. Incentive Compatibility Results

To measure the effect of strategic behavior on the results of the various mechanisms discussed so far, we adapt and build on a process presented by Kominers et al. (2010) and used by Budish and Cantillon (2012). As in those papers, we model student preferences as arising from a combination of personal preferences (simulated randomly for each student as in Section 4) and a common-value component (drawn once for each course in the market). Each student's true preferences are then modeled as the average of the student's personal signal for the value of the course and the market's common-value signal of the value of the course. This leads to a natural mechanism for simulating strategic behavior: a strategic student adjusts the relative weight between the personal signal and the common-value signal, placing greater weight on the common-value signal than on the student's true preferences.

This method for measuring incentives to deviate from truth-telling is an imperfect heuristic, but it gives a realistic basis for comparing various mechanisms in this highly complex environment and is consistent with the notion of bounded rationality. Indeed, in real-life applications, it would be highly *unrealistic* to assume that students would have enough information about the preferences and bids of others to devise an optimal strategic bid. Even if a particular student had *complete and perfect knowledge* of the bids of others (which would be impossible in practice unless the bid submission system was somehow compromised), the problem of devising an optimal strategy under each of the mechanisms in

Figure 3. (Color online) The Average Ordinal Utility per Student per Algorithm, Showing Amounts Above BPM Benchmark with Average Ordinal Utility of 548.15 per Student

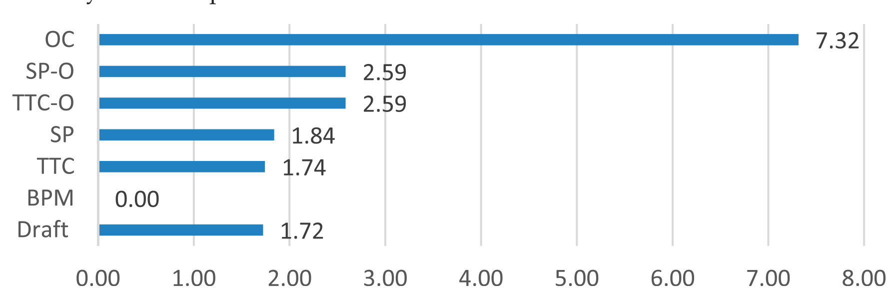
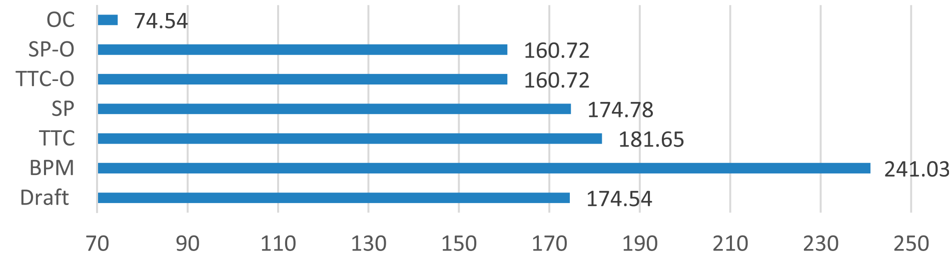


Figure 4. (Color online) The Average Range of Ordinal Utility per Student per Algorithm



question is typically a difficult and large NP-hard optimization problem.

If one instead tried to model beliefs over the preferences of others, the solution concept becomes the determination of a Bayes–Nash equilibrium, which would be even more difficult to compute and again out of the reach of a student in almost any realistic setting. (Although we considered these theoretical problems in our own notes, we leave a detailed study of this problem for future research.) So, instead, we apply the benchmark paradigm explored in the literature, here running the procedure for a broad set of parameter values, including several values for the percentage of students bidding strategically and for how much they manipulate their true preferences. This provides a robust investigation of the possibilities for student gain through deviation from truth-telling.

As in our first set of experiments, we find it beneficial to explore a number of simple metrics to interpret our results. These illustrate how variations in *the percentage of strategic students* and in *the intensity of their strategic behavior* change the outcomes for strategic and truthful students. In this section, we focus on ordinal utility and do not present the parallel results for binary and cardinal utility measures. We do this for brevity and because cardinal results are similar but mildly harder to interpret, and binary utility does not shift drastically based on strategies (a strategic manipulation may shift *which* student gets a small and which a large bundle without changing a binary metric, for example).

To make our heuristic model of strategic behavior precise, consider Equation (20), which is nearly identical to the approach presented in Kominers et al. (2010):

$$Bid_{ij} = (1 - w_i)^* u_{ij} + w_i^* CV_j. \qquad (20)$$

In this equation, Bid_{ij} represents the *declared* cardinal utility of student i for course j, which is calculated as a simple weighted average of student i's personal utility component for course j (labeled u_{ij}) and the common value component of course j in the market (labeled CV_j). The personal parameter w_i controls the relative weighting of these two components. By assumption (as in the previous literature), a bidder who bids truthfully is defined as one who sets $w_i = 0.5$. When student i sets $w_i > 0.5$, student i is effectively pretending to be more influenced by the common-value term than student i's true preferences dictate, causing student i to bid more aggressively on popular courses (those with higher CV_j) and less aggressively on unpopular courses. Such a student is referred to as "strategic." Here, similar to our model in Section 4, students' cardinal and ordinal preferences are forced to be consistent so that a single list of cardinal utility for each course j as in (20) is sufficient to generate a unique ordinal list for each student. As a robustness check, we also explored the settings in which w_i of a truthful student was defined to be 0.25 and 0.75 and repeated all analysis. Our results showed that the conclusions remain largely unchanged for all algorithms. Incentive compatibility metrics of this analysis are discussed in Online Appendix B.

After an initial test of this common-value approach, we noticed systematic poor performance for students using the most straightforward implementation of the approach as described thus far. We then

Figure 5. (Color online) Average Standard Deviation of Ordinal Utility per Student per Algorithm

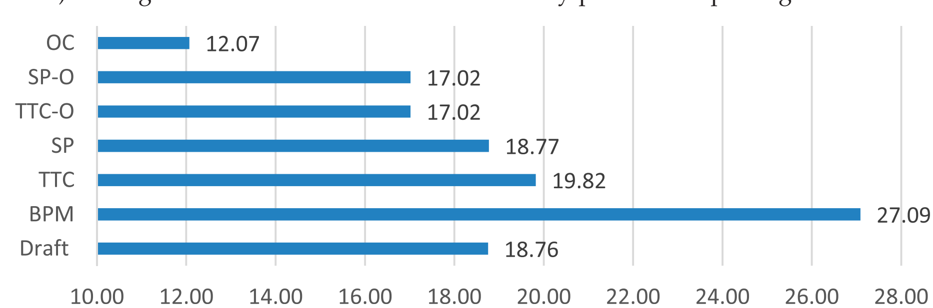
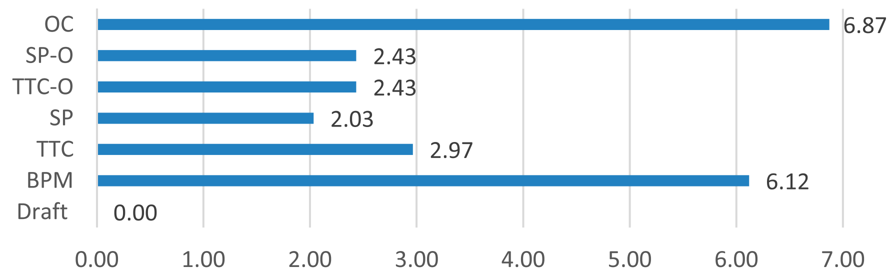


Figure 6. (Color online) The Average Cardinal Utility per Algorithm, Amounts Above Draft Benchmark with Average Cardinal Utility 295.93



found improved payoff to strategic manipulation for students employing the nontruthful common-value adjustment *only* for course sections in the student's "expanded best bundle," defined as the student's favorite feasible bundle together with any course sections with overlap constraints incident to this bundle. Without this measure, many instances saw degraded performance when a high common-value course with low personal value was focused on when an overall better course was available. The results presented here use this heuristically improved method of strategic manipulation.

Given this setup, we varied two parameters to explore the effect of strategic behavior. The first parameter, labeled P, indicates the percentage of students in the market that play strategically. (Hence, $(1-P)$ students bid truthfully, that is, maintaining $w_i = 0.5$.) For each value of P, we also varied the w_i parameter for strategic students. With five values for $P = \{20\%, 40\%, 60\%, 80\%, 100\%\}$ and five values for $w_i = \{0.6, 0.7, 0.8, 0.9, 1.0\}$ of strategic students, each mechanism was tested under 25 different scenarios.

5.1. Effect of w_i When $P = 20\%$

To define the benefit of strategic behavior for each (P, w_i) scenario, we consistently compute the totally truthful benchmark scenario in which $P = 0$. The change in ordinal utility of each strategic student is defined as the total (true) rank of assigned courses to the student in the (P, w_i) scenario minus the total (true) rank in the totally truthful benchmark. The average of this metric over strategic students is shown in Figure 9 for varying w_i values under $P = 20\%$. The left graph in Figure 9 includes the results of all seven algorithms, but to scale better, the right one displays the results of all algorithms except BPM.

Figure 9 illustrates that, as w_i increases, the average benefit of a strategic student decreases in all mechanisms except BPM. It also shows that, among all algorithms, BPM is, by far, the most vulnerable to strategic manipulation with the left graph showing relatively minor differences among all other mechanisms. Zooming in (the right graph), however, does show some subtle differences with the draft mechanism showing the smallest potential benefit to deviation from truth-telling. All of the mechanisms on the right show a decreasing pattern in which larger deviations (via larger w_i values) help the strategic students less, becoming a negative expected value (i.e., the deviation was worse than telling the truth) for a large enough w_i. However, under BPM, strategic behavior never hurts students, indicating that this dishonest strategy is robust.

Overall though, to gauge the magnitude of these effects of strategic manipulation, it is important to remember that ordinal rank measures change rather quickly as the bundle of assigned courses changes. Recall from Section 4.2 that moving from one's best possible schedule to one's second best possible schedule results in a change of six, for example, or that getting one more course (relative to truth-telling) raises ordinal utility by at least 66. Thus, seeing average changes under two ordinal units for each of the mechanisms on the right in Figure 9 indicate that these

Figure 7. (Color online) The Average Range of Cardinal Utility per Algorithm

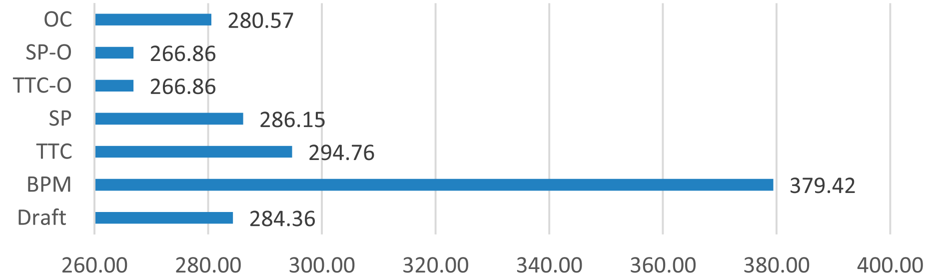
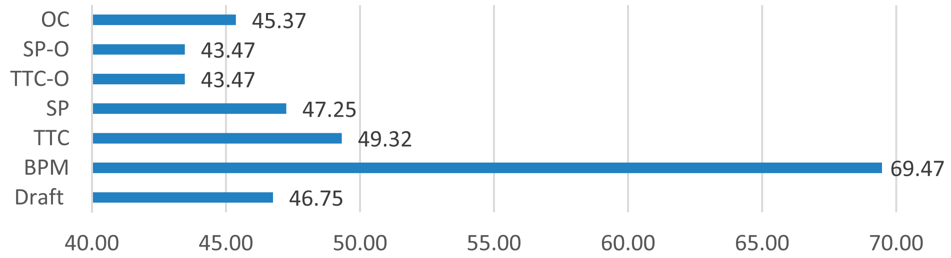


Figure 8. (Color online) Average Standard Deviation of Cardinal Utility per Algorithm



mechanisms are performing rather well in terms of providing very little benefit, on average, to misreporting preferences.

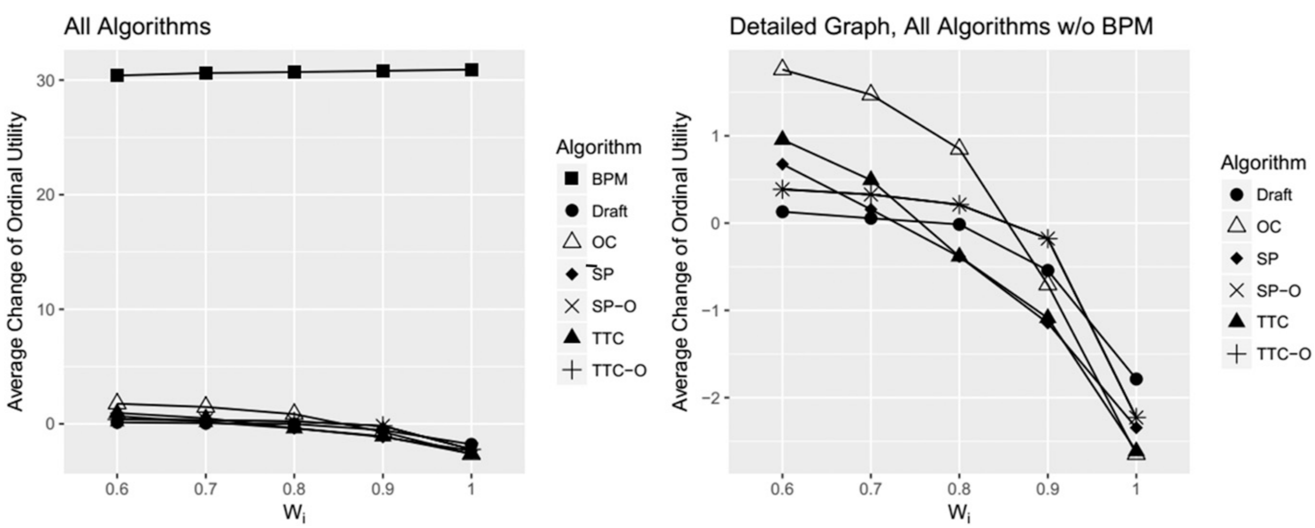
Next, we can measure the average change in ordinal utility between each (P, w_i) scenario and the $P = 0$ benchmark but this time for students who remain truthful in the former case, indicating how much they are harmed by the other students' decision to play strategically.

Figure 10 shows that the strategic behavior in BPM hurts truthful students more than in other mechanisms and that, in the manipulated market, the ordinal utility of the truthful students is between eight and 10 units below their results in the $P = 0$ market. Moreover, in all mechanisms except BPM, the decrease in ordinal utility among truthful students is less than 0.42. For an alternate set of comparisons, we combine the results of Figures 9 and 10 into Figure 11, placing truthful and strategic students side by side for each mechanism.

This figure shows the gap between changes in ordinal utility of strategic students and truthful students in different mechanisms. As might be expected, under BPM, strategic students significantly gain by misreporting their preferences, and their strategic behavior hurts truthful students a bit more than proportionally. By contrast, in TTC-O, SP-O, and the draft mechanisms, the difference between the ordinal utility change of strategic and truthful students is smaller than under other mechanisms. Moreover, among the OC mechanism and other round-by-round mechanisms, OC shows a larger gap between ordinal utility of truthful students and strategic students. However, this gap is small, below 2.5 units in total rank for OC and less than one for all other round-by-round algorithms.

In addition to average effects, the standard deviation of the changes in ordinal utility can be used as a measure of volatility and, therefore, risk. Although the average effect of strategic deviations may be positive in some scenarios for those who play strategically, a large standard deviation around that mean indicates that some will benefit while others lose utility from their decision to play strategically (although those who benefit will outweigh the losers based on the positive average). Figure 12 represents the pattern of changes in standard deviation of ordinal utility of strategic and truthful students. In all cases, we see relatively large standard deviations, indicating that deviation from truth-telling is a risky proposition regardless of the mechanism.

Figure 9. Average Change in Ordinal Utility of Strategic Students Under $P = 20\%$



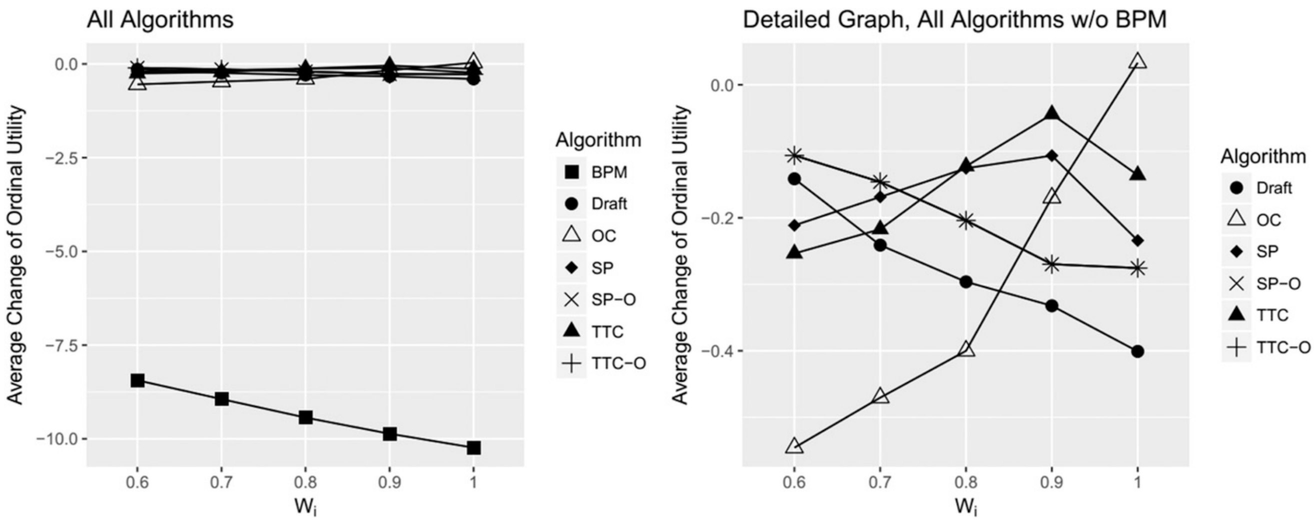


Figure 10. Average Change in Ordinal Utility of Truthful Students Under $P = 20\%$

5.2. Effect of Changing w_i When $P > 20\%$

Here we present an analog to Figure 11 but under $P = 80\%$. Similar results for $P = 40\%, 60\%,$ and 100% are given in Online Appendix B, but the overall story is fairly clear by comparing Figures 11 and 13. As more and more students adopt the non–truth-telling strategy, the benefits of doing so disappear so that the net effect of a large number of strategic students is purely detrimental.

In our further results in Online Appendix B, we see, for example, that when $P = 100\%$, on average, all students would be hurt, and the average change in ordinal utility is negative for all algorithms and all values of w_i, continuing the trend shown here. When everyone deviates from truth-telling, everyone loses.

5.3. Overall Comparisons of Incentive Compatibility

In Section 5.2, we considered the average change in ordinal utility (relative to the $P = 0$ case) for strategic students and truthful students. Let these values (averaged over all students in each group for each (P, w_i) scenario) be denoted ΔOU_{str} and ΔOU_{tr}, respectively. To aggregate the results presented thus far, consider the following metrics. In these three metrics, we did not consider the market with 100%

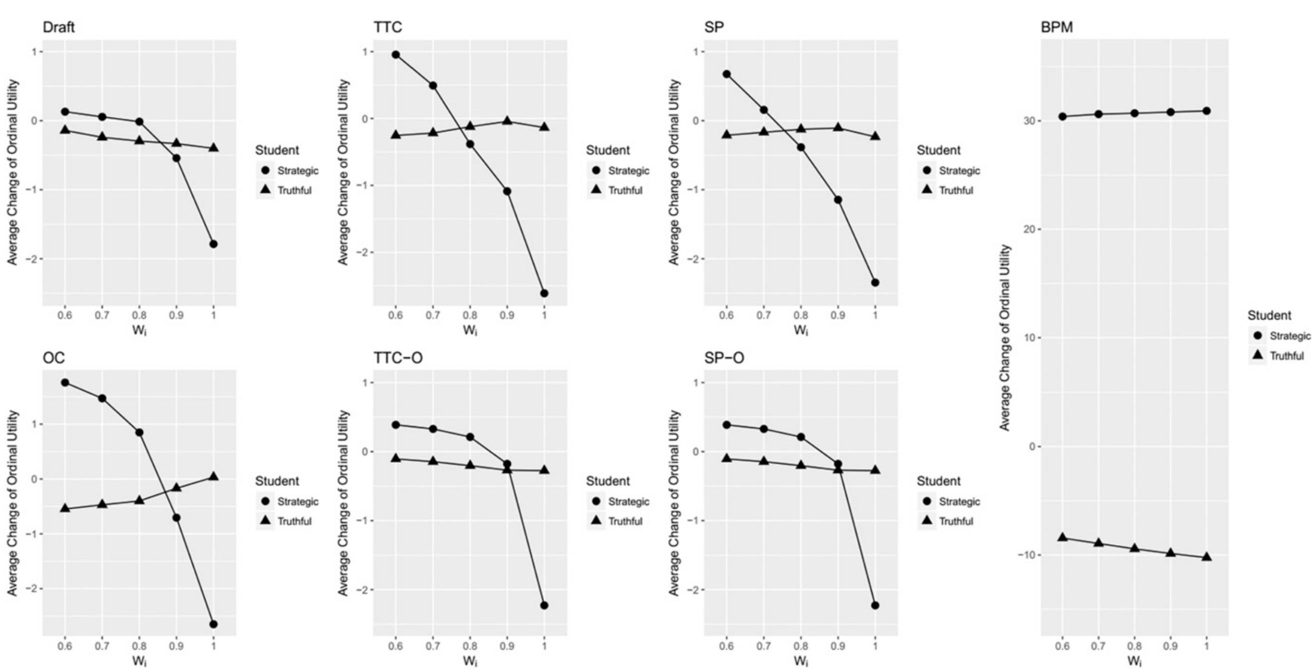
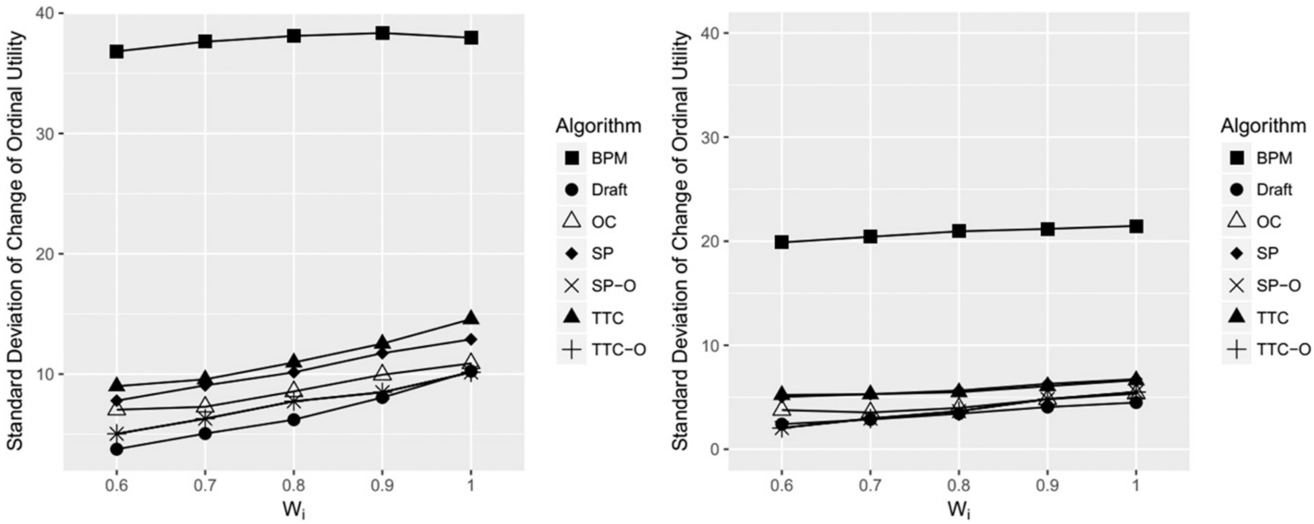


Figure 11. Average Changes in Ordinal Utility of Strategic and Truthful Students Under $P = 20\%$

Figure 12. Left: Standard Deviation of Change in Ordinal Utility of Strategic Students; Right: Standard Deviation of Change in Ordinal Utility of Truthful Students



strategic students because there were no truthful students for comparing with strategic students.

$$\textit{Strategic Upside} = average_{(P,w_i)|\Delta OU_{str}>\Delta OU_{tr}}\ (\Delta OU_{str} - \Delta OU_{tr}),$$
$$\textit{Strategic Downside} = average_{(P,w_i)|\Delta OU_{str}<\Delta OU_{tr}}\ (\Delta OU_{tr} - \Delta OU_{str}),$$
$$\textit{Net Benefit of Strategic Play} = average_{(P,w_i)}\ (\Delta OU_{str} - \Delta OU_{tr}).$$

When the net benefit of strategic play gives a neutral aggregate measure of the average marginal ordinal utility of changing from a truthful student to a strategic student, when the scenario is unknown, the upside and downside of strategic play are also informative in case the downside of a decision is weighted more heavily than the upside (as in loss aversion; see Kahneman and Tversky 1984).

Comparing Figures 14, 15, and 16, we see a few overall conclusions. BPM tends to offer a robust reward to those who play strategically, on average, with no downside to doing so. The net benefit to strategic play is negative for all other mechanisms except for OC, with which still it is quite close to zero. Overall, strategic play does not tend to help very much in these

Figure 13. Average Changes in Ordinal Utility of Strategic and Truthful Students Under $P = 80\%$

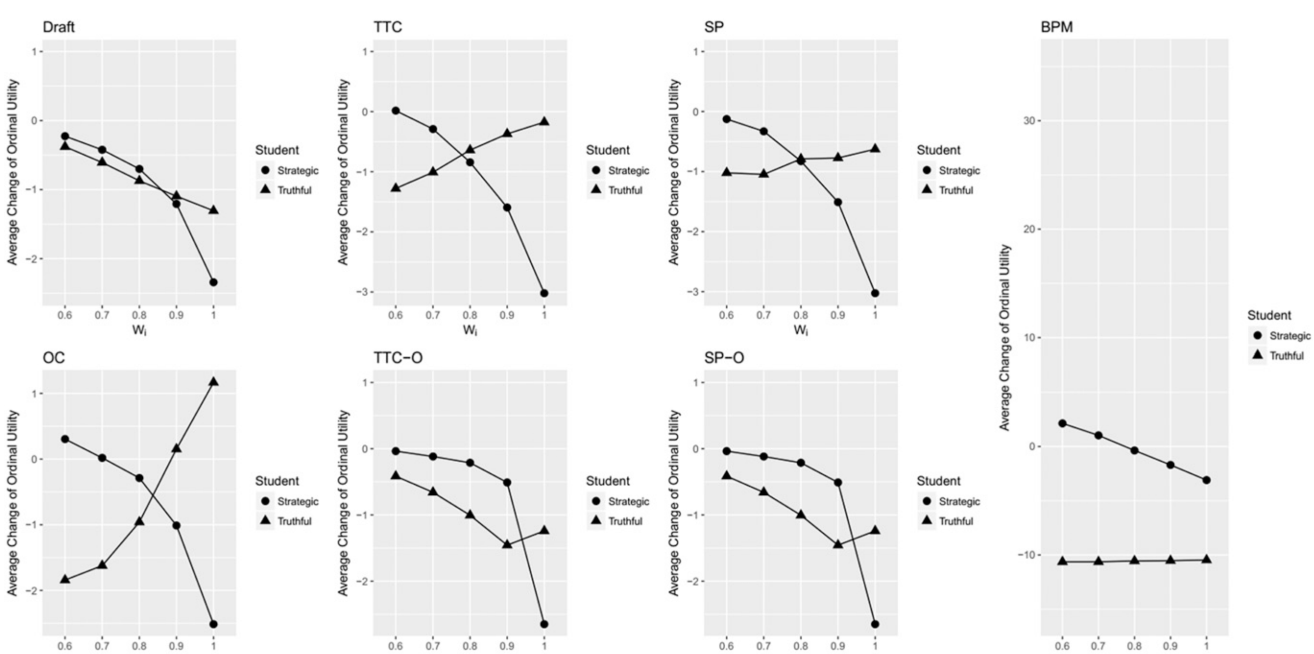


Figure 14. (Color online) Strategic Upside (Ordinal Utility)

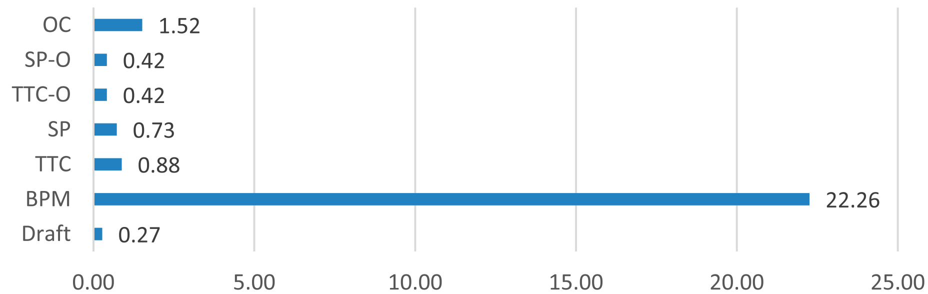
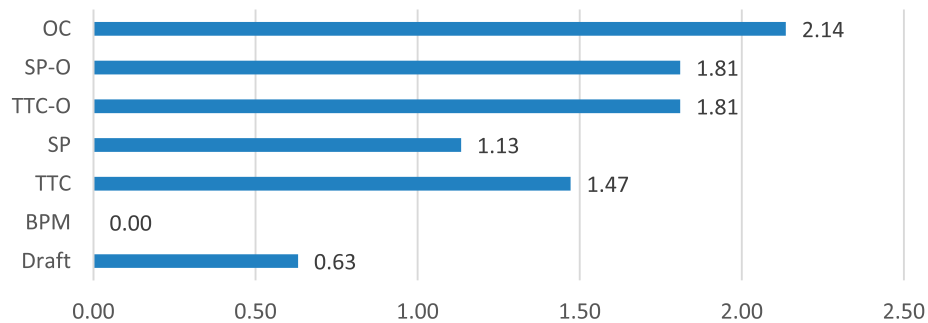


Figure 15. (Color online) Strategic Downside (Ordinal Utility)



latter (non-BPM) mechanisms. For OC, although there is a tiny benefit, on average, in the net, strategic play is more volatile, introducing the chance of a larger downside than any other mechanism. The existence of this larger downside risk may, in itself, be viewed as a deterrent to strategic play.

6. Discussion and Conclusion

This paper has explored five new algorithmic variations to solve the course-allocation problem, addressing the relative balance of efficiency, fairness, and incentive compatibility. To understand the performance of these five algorithms, we compared them with the draft and bidding-point mechanisms, benchmarks that have been used in practice. We introduced a comprehensive systemization of natural metrics for these objectives with a novel perspective of analyzing any set of outcomes based on the total (or average), range, and standard deviation of binary, ordinal, and cardinal utility.

As a final comparison of the mechanisms in question, consider the following selected graphs based on the relative rankings (higher being better) of the seven mechanisms for CAP.

The clearest ordering is in fairness, shown in Figure 17, in which OC dominates whether one looks at binary or ordinal efficiency. Recall that SP-O and TTC-O switch places with OC when considering cardinal fairness but with only a small lead compared with OC's large binary and ordinal lead shown in Figures 2, 4, and 5. Turning our attention to efficiency in ordinal and binary utility, the results remain essentially the same but with a murkier comparison among SP, TTC, and draft as in Figure 18.

Looking at incentive compatibility, the results are less clear. In Figure 19, we see that comparisons based

Figure 16. (Color online) The Average Net Benefit of Strategic Play (Ordinal Utility)

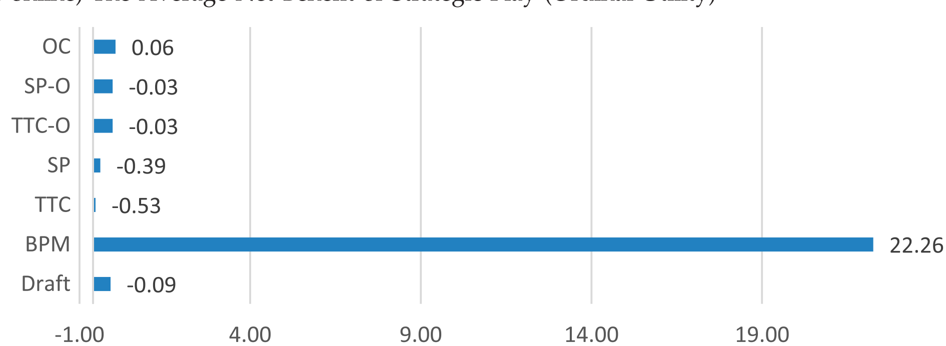
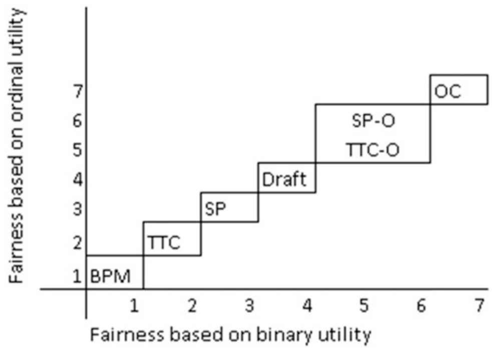


Figure 17. Ranking the Algorithms Based on Fairness (Standard Deviation or Range)



on upside opportunities (smaller gains representing less temptation) and downside risks (larger downsides representing a larger deterrent) are less consistent. For example, the draft performs well in not providing large opportunities for upside, and OC performs well in deterring strategic play based on the downside risk of unsuccessful manipulation.

Still, perhaps the main conclusion to derive from our experiments on incentives for strategic play is that, in the net, except for BPM, all mechanisms provide very little benefit to manipulation (see Figure 16) with the risks of manipulation almost always (at least nearly) outweighing the potential benefits. Indeed, given the nearly universal bad results for BPM, we are inclined to remove it from consideration entirely and consider a comparison among the other mechanisms. The remaining mechanisms perform quite similarly on incentives, leading us to consider a ranking on efficiency and fairness metrics, in which case OC is a clear "winner" among the mechanisms investigated here.

If there were reason to believe that incentives were an issue, SP-O or TTC-O seems to provide slightly better incentives than OC with the smallest degradation on the other metrics overall. In the numerical simulations conducted here, SP-O and TTC-O performed nearly identically, perhaps because of the amount of competition in the large-scale market although small

Figure 18. Ranking the Algorithms Based on Binary and Ordinal Efficiency

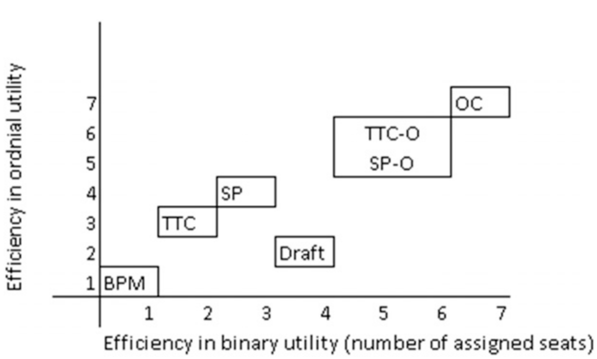
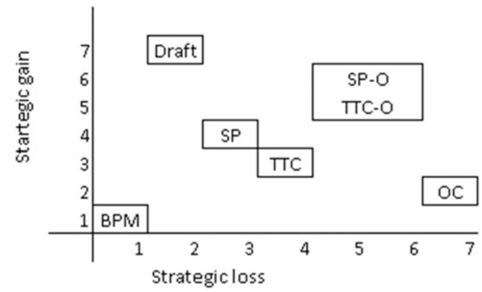


Figure 19. Ranking the Algorithms Based on Incentive Compatibility



examples can be devised in which SP-O's use of second prices removes some opportunities for manipulation. This tends to give SP-O an ever-so-slight edge over TTC-O but one that is nearly negligible. Overall, we hope to have shown the opportunity for a significant improvement in the performance of course-selection algorithms through the direct application of optimization, whether on a round-by-round or market-wide level.

Finally, our additional experiments shown in Online Appendix D indicate that SP-O and TTC-O maintain better fairness properties in cases of extreme correlation as opposed to OC, which will instead sacrifice fairness for efficiency in such circumstances. But these extreme cases seem less likely in practice, and our results indicate very little hope for any mechanism in such cases. If everyone is in complete agreement, trying to use a mechanism to get them to admit where their preferences are different is futile. Still, these additional results point to the possibility of strengthening the OC algorithm through additional fairness constraints and reinforce the necessity of considering round-by-round mechanisms such as SP-O and TTC-O in discussions of future applications.

Endnote

[1] See http://www.hbs.edu/mba/registrar/crossregistration/Documents/fsched13.pdf and http://www.hbs.edu/mba/registrar/crossregistration/Documents/wsched14.pdf.

References

Akbarpour M, Nikzad A (2019) Approximate random allocation mechanisms. Working paper, Stanford University, Stanford, CA.

Budish E (2011) The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *J. Political Econom.* 119(6):1061–1103.

Budish E, Cantillon E (2012) The multi-unit assignment problem: Theory and evidence from course allocation at Harvard. *Amer. Econom. Rev.* 102(5):2237–2271.

Budish E, Che Y, Kojima F, Milgrom P (2013) Designing random allocation mechanisms: Theory and applications. *Amer. Econom. Rev.* 103(2):585–623.

Diebold F, Aziz H, Bichler M, Matthes F, Schneider A (2014) Course allocation via stable matching. *Bus. Inform. Systems Engrg.* 6(2): 97–110.

Ehlers L, Klaus B (2003) Coalitional strategy-proof and resource-monotonic solutions for multiple assignment problems. *Soc. Choice Welfare* 21(2):265–280.

Hatfield J (2009) Strategy-proof, efficient, and nonbossy quota allocations. *Soc. Choice Welfare* 33(3):505–515.

Kahneman D, Tversky A (1984) Choices, values, and frames. *Amer. Psychologist* 39(4):341–350.

Kominers SD, Ruberry M, Ullman J (2010) Course allocation by proxy auction. Saberi A, ed. *6th Internat. Workshop Internet Network Econom.*, Lecture Notes in Computer Science, vol. 6484 (Springer, Berlin, Heidelberg), 551–558.

Krishna A, Ünver U (2008) Improving the efficiency of course bidding at business schools: Field and laboratory studies. *Marketing Sci.* 27(2):262–282.

Morrill T (2012) Making efficient school assignment fairer. Working paper, North Carolina State University, Raleigh.

Nguyen T, Peivandi A, and Vohra R (2014) One-sided matching with limited complementarities. Working Paper No. 14-030, Penn Institute for Economic Research, University of Pennsylvania, Philadelphia.

Nogareda A, Camacho D (2016) Optimizing satisfaction in a multi-courses allocation problem. Novais P, Camacho D, Analide C, Seghrouchni AEF, Badica C, eds. *Intelligent Distributed Computing IX, Proc. 9th Internat. Symp. Intelligent Distributed Comput.*, Studies in Computational Intelligence, vol. 616 (Springer, Cham, Switzerland), 247–256.

Papai S (2001) Strategyproof and nonbossy multiple assignments. *J. Public Econom. Theory* 3(3):257–271.

Roth A, Sotomayor M (1990) *Two-Sided Matching* (Cambridge University Press, Cambridge, UK).

Schummer J, Vohra RV (2007) Mechanism design without money. Nisan N, Roughgarden T, Tardos E, Vazirani V, eds. *Algorithmic Game Theory* (Cambridge University Press, Cambridge, UK), 243–265.

Sönmez T, Ünver U (2010) Course bidding at business schools. *Internat. Econom. Rev.* 51(1):99–123.