8-2020

Allocation of Public Resources: Bringing Order to Chaos

Lance Clifner
Chapman University, clifn100@mail.chapman.edu

.

Allocation of Public Resources: Bringing Order to Chaos

A Dissertation by

Lance Clifner

Chapman University

Orange, CA

Schmid College of Science and Technology

Submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Computational and Data Sciences

August 2020

Committee in charge:

Stephen Rassenti, Ph.D., Chair

David Porter, Ph.D.

Erik Linstead, Ph.D.

CHAPMAN UNIVERSITY
SCHMID COLLEGE OF SCIENCE AND TECHNOLOGY
Computational and Data Sciences

The dissertation of Lance Clifner is approved.

Stephen Rassenti, Ph.D., Chair

David Porter, Ph.D.

Erik Linstead, Ph.D.

May 2020

Allocation of Public Resources: Bringing Order to Chaos

by Lance Clifner

ACKNOWLEDGEMENTS

- Dr. Dave Porter, PhD – for taking a chance on me 26 years ago to do the software on a commercial version of a cap and trade emissions auction system and introducing me to experimental economics and combinatorics.
- Dr. Steve Rassenti, PhD – for believing in me and inviting me to work on numerous very sophisticated experimental economics projects over the last two decades.
- Dr. Erik Linstead – for introducing me to data mining and reinforcing the idea to always ask the question: "but does it make sense?"
- Economic Science Institute – for providing tuition support, research opportunities, administrative and moral support over the entire duration of this project.
- Kay Family Foundation – for providing financial support in my final year of this project, allowing me to focus on adding elective course assignment and not have to do part-time work to keep the mortgage paid.
- Dr. Joan Chen, PhD – for being the best Science Olympiad assistant head coach for whom one could ask and being the person who, unintentionally, is responsible for launching me on this 7 year adventure.
- Dr. Irina Kufareva, PhD – for being the originator of the crowdsource team selection idea, a stickler on student performance data collection, handling, and analysis, and her considered and considerable input and feedback on the team selection algorithm, and continued support of and coaching students in SO even though her kids are long gone from the SO program.
- Dr. Eugene Rausch, PhD – for programming the original greedy team selection algorithm, providing the computational power (1000 CPUs!) to run the team selection algorithm over a 5 year period, programming the crowdsource team selection web page and hosting the crowdsource page for all 7 years of this project.
- Mike Holmes, JD – for his consistent and measured advice on organizational structure, rules, and procedures, and his always level-headed advice and point of view on volatile topics.
- Dr. Scott Wooden, PhD – for taking over as SO head coach and easily demonstrating that the institution we designed and implemented works smoothly with a rational person leading it.
- Liz Jablecki – for her 25+ years of practical experience with SO both in running a parent-lead program at a school and running the San Diego Regional SO Competition, acting as a sound board for our ideas on re-vamping our school's SO program and providing advice on handling parents and students in so many different situations.
- Zoom Duong – for spending 5 hours one afternoon talking with Joan and me about SO, on ideas and her practical experience on how to organize and run a school's SO program, and the motto "Head Coach's decision is final."

- Jim Byers and Sunset Software – for providing (for free!) the XA linear programming solver used in the second generation team selection algorithm, and for his quick assistance, guidance, and programming the initial IP version of team selection.
- The gentlemen in the Kaiser Prostate Cancer Support Group at Garfield for helping to educate me after my cancer diagnosis, to become my own health advocate, and, most importantly for putting my situation into perspective and getting my life back to 'normal'.
- Stephanie Clifner – for automating the Event assignment on the SOTM website (freeing up a lot of my time no longer having to run Event assignment for the schools manually) as well as adding many other new features and fixing numerous pre-existing bugs.
- Benjamin Clifner – for assisting with organizing, filtering, analyzing, plotting the student performance data, and arguing about the right way to do statistics and data analysis.
- Lisa Clifner – for allowing me to pursue, and supporting me in, this 4-year, part-time sprint that turned into a 7-year, full-time marathon and personal obsession.

ABSTRACT

Allocation of Public Resources:
Bringing Order to Chaos

by Lance Clifner

Science Olympiad (SO) is a team-based academic competition involving multiple subject areas (Events) with arcane rules governing the team composition. Add to the mix parental contention over which student(s) get on the "All-Star" team, and you have a potentially explosive situation. This project brings order and logic to school-based SO programs and defuses tense milestones through the implementation of an institutional structure that: assigns students to Events based on solicited student preferences for the Events, collects objective student performance data, composes competitive teams based on student performance (aka "Moneyball"), and brings transparency to the Team Selection process through crowdsourcing. The Event Assignment mechanism is simple, fast, easy to understand, and yields Pareto-optimal results based on student preferences, without the exchange of money or tokens, and with effectively no incentive to game the system. The Team Selection mechanism optimizes student performance data from teachers (Event Coaches) and competitions to compose a tiered series of teams with the greatest potential performance. And the Crowdsource Tool allows any stakeholder to compose a candidate team for advancing to the State competition, where the team with the highest potential performance score advances to State whether the team was composed with the Crowdsource Tool or by the Team Selection algorithm. The end result is that students get more of the Events that they want; Team Selection is transparent and far less contentious; teams are higher quality; and managing the SO program for a school takes considerably less time and effort.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Term	Definition
DGS	**D**emange, **G**ale, and **S**otomayor auction.
EA	**E**vent **A**ssignment
Event	Equivalent to a course offered at a school.
FCFS	**F**irst-**C**ome, **F**irst-**S**erved
HBS	**H**arvard **B**usiness **S**chool
IP	**I**nteger **P**rogram
ITC	International Timetabling Competition. https://www.itc2019.org/home
LP	**L**inear **P**rogram
MI	**M**isery **I**ndex (see section 3.1)
MV	**M**isery **V**alue (see section 3.1)
NSAP	**N**ew **S**tudent **A**llocation **P**roblem
R+S	**R**egional and **S**tate
RSD	**R**andom **S**erial **D**ictator
SAP	**S**pace **A**llocation **P**roblem
SD	**S**erial **D**ictator
SO Season	An **SO** season runs in sync with the standard school year. Thus the 2015 SO season runs from the fall of 2014 through the spring of 2015.
SOP	**S**cience **O**lympiad **P**rogram. This is the organization that manages Science Olympiad at a school.
SOTM	**S**cience **O**lympiad **T**eam **M**anagement software/website. A website built during this project to assist with the management of students and teams in an SO program a school site.
SP	**S**trategy **P**roof
SPAP	**S**tudent-**P**roject **A**llocation **P**roblem.
TS	**T**eam **S**election

Term	Definition
TSIP	**T**eam **S**election **I**nteger **P**rogram
UMBS	Ross **S**chool of **B**usiness at the **U**niversity of **M**ichigan. Progenitor of the UMBS course bidding system used at a number of universities around the United States.
VB	**V**isual **B**asic
VL	**V**ickrey-**L**eonard Auction

Allocation of Public Resources: Bringing Order to Chaos

Computational and Data Sciences
Economic Science Institute
Chapman University
by Lance Clifner
May 2020

1 Introduction/Background

The Assignment Problem is one of the most studied problems in the fields of economics, operations research, and computer science. It covers a wide variety of domains including these examples:

- scheduling[6] – assigning students to courses
- trading[30] – thin market trading of emissions credits
- transportation[31][33] – auctioning and matching trucks, lanes, full-load and less-than-load cargos; auctioning airport take-off and landing slots combinatorically
- workflow[34] – assigning tasks to workers
- sports[21][24] – cricket team selection using multi-objective optimization; team selection for Fantasy Football (both American Football and International Football)
- resource allocation[32] – allocating mass and power on space probes

A common approach to solving these types of problems is some form of combinatorial optimization algorithm.

In concept, the Assignment Problem involves the process of optimally assigning *n* objects in one set to *m* objects in a different set. This task is accomplished subject to a set of assignment constraints that limit the ways in which the assignment is allowed to be made, and an objective function that measures the value of an assignment. The assignment constraints are used to encode the goals to be achieved and restrictions to be followed. The objective function measures how effective any particular assignment is.

In the education domain, the problem of assigning students, courses, teachers, facilities, and timeslots can be divided into two main considerations: timetabling and allocation. Timetabling concerns setting the course schedule by assigning teachers to the courses and timeslots, and allocating facilities such as classrooms and labs. Allocation concerns setting student schedules by assigning students to the timetabled courses. Both timetabling and allocation involve various constraints on permissible assignments. For example, two courses can't be taught in the same classroom in the same timeslot, and one student can't be in two different courses in the same timeslot.

In the realm of team sports, using 'Moneyball'[1]-style [25][26] strategies to compose an entire team combinatorically can yield a potentially higher performing team than a team formed

[1] Moneyball is a book and a movie about a real-life Event with the Oakland Athletics professional baseball team in the early 2000's, where the team recruited their players based on objective statistics with the intent to find under-valued players. The Oakland A's were able to form a highly competitive team on a relatively shoe-string budget. It

through typical subjective measures[2]. The more complicated the interrelationship of team members and the rules and requirements of team composition, the more critically important an optimizing algorithm can be to building the best team. This task of selecting a holistic team rather than the greedy selection of individuals brings not only the different skill sets of players into consideration when forming a team, but also objective measures of proficiency in those skills. These players each have varying skills and skill levels and the multi-player team *requires* a specific overall mix of skills in order to be competitive. For example, on an American football team, which has multiple different skilled positions, having the 11 best quarterbacks in the league probably won't make for a very competitive team against another team with 11 above-average players who each are skilled in one (or more) of the 11 positions on a team.

1.1 Introduction to Science Olympiad

Science Olympiad™ (SO) is a STEM competition held annually amongst teams of students from over 7,800 elementary, middle, and high schools across the United States.[3] The teams progress through an elimination series of Regional, State, and National Competitions. The top team from each division is invited to the White House to meet with the President.

Each team is composed of 15 students, with the students on a single team competing in pairs or triples on a set of 23 STEM-related Events. Each student on a team competes in 2 to 5 Events, with the typical student competing in 3 Events, and a few students competing in 4 Events. These are some of the rather arcane set of team configuration rules for an SO team—and they are surprisingly complicated for a competition targeted at middle and high school students.[4] These overtly complicated rules make SO fertile ground for an algorithmic and combinatorial approach for Event assignment and team selection.

Regional Competitions may have 30 to upwards of 100 teams competing, with 8 to 10 schools each advancing a single team to the State Competition. One team advances from each State to the National Competition.

At the San Diego Regional Competition, a school may enter up to 6 teams into the competition, with the highest-scoring team from that school determining the school's overall placement in the Regional Competition. A school in the San Diego Regional competition may have up to 250 students in their school's SO program. However, at most, only 90 students[5] from one school may compete in the San Diego Regional Competition.

1.2 Introduction to Event Assignment

The process of assigning students to their preferred Events in SO fits neatly into the category of Generalized Assignment Problems under Educational Allocation problems in the category of

---

should be noted that the previous recruiting method involved many subjective measures, actually including "measures" such as how "hot" a pitcher's girlfriend was. Now, many professional sports teams around the world have adopted similar objective strategies for building their teams.

[2] This is especially true when the traditional methods involve picking one team member at a time.

[3] https://www.soinc.org/info/about-science-olympiad - more information about SO can be found here.

[4] For example, a simpler team configuration might be: 16 students on a team, 24 Events, 2 students per Event. Thus making for 2 student-seats/Event * 24 Events/team = 48 student-seats/team. Which means that each student participates in 3 Events (48 student-seats/team / 16 students/team = 3 seats per student ➔ 3 Events per student). But, with 23 Events, 2 or 3 students per Event, and 15 students per team, these parameters do not allow for nice round numbers. The complete set of team configuration rules is found in Appendix A.

[5] Maximum 6 teams with a maximum of 15 students per team: 6*15 = 90.

Student-Project Allocation Problems (SPAP). This Event Assignment process is one topic in this dissertation. The Timetabling aspect is not addressed in this dissertation since the schedule availability of the parent coaches was almost always a single timeslot, and their Event of interest was almost always a single Event: thus, the resulting timetable was essentially fixed due to parental constraints.

1.3 Introduction to Team Selection

The process of selecting Teams based on student performance in various Events is also a form of the Generalized Assignment Problem in the arena of combinatorics similar to scoring and selecting teams for Fantasy Football[21] and major league professional sports[22][26]. Because of the intrinsic involvement of parents coaching teams in which their or their neighbors' children are involved, in SO schools with multiple teams, it is imperative that the team selection process be guided by a well-defined mathematical objective function. The optimized, performance-based Team Selection process is extremely helpful in keeping peace amongst the parental units. This Team Selection process is another topic of this dissertation.

2 Related Literature

There is a mass of literature on course selection for students based on student preferences and team selection based on player performance. A person could make a career out of reading and reviewing the currently available literature. Much of the literature is repetitive with minor variations in application. However, there are a few themes that thread through the literature which are relevant to this project.

2.1 Course Assignment

Course Assignment involves far more than simply creating student schedules by assigning students to courses. In order to create a student schedule, a master schedule of when and where courses are offered and who is teaching them needs to be known. In order to create a master schedule, it is necessary to know which courses are going to be offered, the student demand for those courses, what facilities are available for conducting classes, which courses require special equipment, which teachers are capable of instructing the courses and which courses the teachers are interested in teaching, among many other pieces of information.

2.1.1 Timetabling and Course Assignment Tasks

Timetabling and Course Assignment are typically handled independently.[10] The separation of these two tasks makes each task far simpler to perform, and makes the total problem of timetabling and assignment much easier than if the two tasks were performed simultaneously. The cost of separating these tasks is that the overall efficiency of the assignment result will be no greater than (and almost always guaranteed to be less than) the efficiency of a system which tackled both tasks simultaneously.

The timetabling task is frequently done first to set the course availability schedule. As described by Burke et al.[11] "a timetabling problem is a problem with four parameters: T, a finite set of times; R, a finite set of resources; M, a finite set of meetings; and C, a finite set of constraints. The problem is to assign time and resources to the meetings so as to satisfy the constraints as much as possible." In their review of timetabling methodologies and constraints, Faudzi et al.[10] asserted that timetabling can be divided into three problem types: examination timetabling[6], course timetabling[7], and school timetabling[8]. Faudzi further shows that every solution finding and optimization method known to man has been applied to timetabling: exact (optimal), heuristic, genetic algorithm, local search, and various hybridizations, etc.

With a known schedule, students can then be assigned to courses[9]. Here, Faudzi et al. [10] divide the allocation problem into three categories: student-project allocation problem[10] (SPAP), new student allocation problem[11] (NSAP), and space allocation problem[12] (SAP).

Given the computing power available today and the speed of current linear solving software, there is no reason to think that these tasks cannot be combined into a single, optimized operation—however, this unified approach to scheduling is a topic for a future project. In fact, there is an International Timetabling Competition[13] (ITC) held annually wherein teams compete to see who can come up with the best university class schedule complete with rosters of students. Curiously among the many, many constraints in the ITC, students are not allowed to have any course preferences (but students are required to be slotted into courses)[14] nor are instructors included anywhere in the constraints or data.

2.1.2 Preference-Based Assignment

According to Faudzi et al.'s survey [10] of SPAP, both teachers and students are likely to have preferences for which courses they are assigned. Teachers certainly have preferences for what they teach and when they teach it. Students may have preferences (but may not be allowed to express those preferences) for which courses they take and when they take courses. And teachers may have preferences towards students, which may be incorporated or excluded from the assignment model.

Preference-based assignment is usually expressed in one of two ways:

- Ordinal Preferences – this method can't signal the intensity of one ordinal preference over an adjacent ordinal preference, but the agent will be assigned things in the ordinal order in which the agent specified them. For example, the agent won't be assigned their 3rd most preferred item unless their 1st and 2nd items were not available.

[6] This is effectively setting the finals week schedule at the end of a term.
[7] This is setting a course schedule, including meeting times for courses and sessions, room assignments, and teacher assignments.
[8] This is an inverse working of course timetabling, where students are pre-assigned to courses and sessions and then teachers and rooms are assigned. This results in an NP-hard or NP-complete problem.[10][7]
[9] For a course assignment problem of even mild complexity it should be noted that performing these two tasks in the opposite sequence makes the overall problem intractable. That is, setting up a schedule after students are assigned to courses with no regard for schedule time slots, teachers, and classroom availability, leaves such an entangled web that a schedule cannot be teased out to meet the constraints of time slots, teachers, and classrooms without having to move students around in some of their course assignments.
[10] Assigning students to courses based on student preferences. More generally, students are assigned to projects based on student preferences *and* project leader preferences.
[11] New students are assigned to courses based on ability level to match the current ability level in those course sessions.
[12] The goal of SAP is to maximize the efficient use of space. To me, this seems to belong more in the timetabling problem category.
[13] https://www.itc2019.org/home
[14] https://www.itc2019.org/format#students

- Intensity Preferences – using money or tokens to indicate the degree to which one thing is valued relative to another, but the agent may end up getting less-preferred things in an ordinal sense. For example, the agent *could* be assigned their 3rd most preferred item if the agent bid more tokens for their 3rd most preferred than were bid on their 1st and 2nd items.

Some have tried using both Ordinal and Intensity preferences, such as Sonmez & Unver with their modified Gayle-Shapley assignment mechanism [13], but this could only be done in the laboratory and they were not able to try it in the field. Others have tried implementing combinatorial preference entry, but this has proven to be extremely complex and difficult—even with highly sophisticated subjects with recent domain expertise, on average, fewer than 25% of the subjects even *attempted* combinatorial order entry. [29]

2.1.3 Indicating Preferences with Money or Tokens

As concluded by Sonmez & Unver[13], using tokens (bidding) "can result in allocations that are not market outcomes and result in unnecessary loss of efficiency." This is at least in part "due to the two possibly conflicting roles of the student bids[15]." In addition, by collecting both ordinal rankings of courses *and* bids for those courses, Sonmez claims that using a Gayle-Shapley assignment mechanism "has the potential to make a substantially larger proportion of students better off (approximately 20% in our study)" than the current UMBS bid-only (token) system.

Also, as seen in the analysis of Sonmez & Unver [13] (271), with the UMBS bid-only system, it is not unusual for a student to end up being assigned to a popular course for which they bid high. But that *popular* course was actually *lower* on the student's *ordinal* preference list than other (less popular) courses(s) which they were *not* assigned because the student bid *lower* on those courses—specifically because they were less popular. Thus, a big flaw in the UMBS system is that many students end up getting *ordinally* less preferred courses because of the design of the bidding system.

As described by Olson & Porter [12], token-based auctions "do not elicit more information about an agent's relative valuations than just ordinal rankings." And in their Result 6: "In the high contention environment, the non-transfer schemes (SD, Chits[16]) provide significantly more surplus to agents than the auctions (VL, DGS[17])." However, Olson & Porter also contend that the non-transfer mechanisms were less efficient than the transfer mechanisms. In low contention environments, the mechanism differences for surplus and efficiency were small. In high contention environments, the *non-transfer* mechanisms yielded up to 300% *higher* surplus for the *agents* with at most a 10% *lower* efficiency (89% vs. 99%). Olson & Porter also concluded if "one is restricted to non-transfer mechanisms then the use of chits provides better efficiency outcomes than ordinal rank mechanisms but it is hardly impressive."

[15] "Many schools use bidding systems in which student bids are used both to infer preferences over courses and to determine student priorities for courses." [13] However, when this study solicited ordinal ranking information from the students to put their bids in context, they found that with a large portion (77%) of the students their monotonic bid order for courses did *not* match their ordinal preference for courses, meaning that they "were gaming the system."

[16] SD – Serial Dictator. Chits – bidding with tokens (tokens have no value outside of the auction and cannot be carried over from one auction to the next). These are mechanisms where nothing of value is exchanged between the agents involved.

[17] VL – Vickrey-Leonard auction. DGS – Demange, Gale, and Sotomayor auction. These are mechanisms where money is exchanged between the agents and/or auction conductor.

So, while bidding with money may provide more *system* surplus than bidding with tokens, using money to bid for courses in the California public school system is simply not an option[18]. And since tokens would require more sophistication from the student, but do not provide appreciably better results, ordinal ranking is more than sufficient for indicating course preferences for this project.

2.1.4 Serial Assignment Mechanisms

As shown by Olson & Porter [12] and as discussed by Bade [11] and Budish [7], serial assignment mechanisms have reasonably high efficiencies for such a simple and easy to explain mechanism. In single round serial mechanisms, the agents are always best off revealing their true values [16]. In multi-round serial mechanisms, it has been shown that agents *may* be better off if they swap their order of preferences, placing more popular (but their less-preferred) items higher in their preference list, and moving less popular (but their more-preferred) items lower in their preference list [6][7][13]. The trade-off from these types of gaming attempts is that the agents may end up with their swapped (and lower preferred) items and miss out on their more highly preferred (but artificially lower-ranked) items.

Note that gaming is not limited to serial mechanisms, but is also present in bidding systems. [13]

2.1.5 Other

In the Faudzi et al survey [10], they list the constraints commonly used for SPAP as preference lists, capacity constraints, and one course per student per timeslot.

Budish [6](32) measures student success by using the average ordinal rank of the classes assigned to the students. This mirrors the use of Misery Index (see section 3.1) in this project. Their finding, albeit with inferred student preferences applied to incomplete data, is that students are much better off being truthful in their preferences than in trying to game the system. The very, very small number of students (less than 0.5% out of 10,000) who fared better by being strategic with their preferences were those students who "mainly liked unpopular courses" and likely sought only one popular course.

Note that Budish [6] (32) infers truthful student preferences vs. gaming activity based on the same survey taken at different points in time. The assumption is that students truthfully reveal their preference for courses in May (when their preferences won't actually be used for assignment), and that strategic preferences are revealed in July (when their preferences *are* used for assignment).

2.2 Team Selection

This area is also widely studied, with earlier work concentrating on project teams and other groups of people, and only more recently with a plethora of work on sports teams. With regard to competitive team performance, with only the exception of the Oakland Athletics, all of these papers are theoretical in that their algorithms and results cannot actually be tested out in the real world to see if there is an impact on performance. That is to say, the papers only create 'fantasy' teams from players from multiple teams within a league, where those players are never gathered together on an actual team to see how they would perform as a team against other teams. In the world of professional sports, where players are recruited based on performance metrics as a

[18] In fact, it is illegal to require students pay money to obtain courses or materials in the California public school system.

result of Moneyball, the algorithms and systems are proprietary and private; for financial and competitive reasons, these professional systems are not divulged and thus cannot be compared either.

2.2.1 Moneyball

Have traditional methods of selecting players for professional teams been wrong? The short answer, according to Hakes [25] is 'yes'. Not only does the skill set of the player matter, but also the level of their expertise with their skills. In addition, due to team salary caps, being able to figure out which players together provide the biggest bang-for-the-bucks is critical for a competitive edge. [26] [25]

2.2.2 Performance Based

It's important to have not only a player's performance statistics and accurate measures of those statistics, but to also know which statistics are most relevant with regard to team performance. For example, Ahmed et al [21] used multiple player statistics and player position to select cricket teams from players within a league, and analyzed the statistics that appeared to be most important to identify high performing players. Whereas Omkar [27] used players' win/loss records to pick players for a cricket team, which invites the question of free-riders: just because a player is on a winning (or losing) team doesn't mean that that specific player is actually a high (or low) performing player.

2.2.3 Optimization Methods

As one might expect from the variety of optimization methods used in course assignment, there are as many optimization methods used to perform team selection as there are different optimization methods. Similar to course assignment, the most common method is linear optimization [21] (whether single or multi-objective), also used are genetic algorithms [27], TOPSIS [23], fuzzy logic, greedy, and many more.

2.2.4 Bidding Prediction Systems

As shown by Ryan [24] and Anagnostopoulos [22], bidding prediction systems work well in situations where two or more agents are competing with each other in forming teams. In situations where there is only one agent creating teams from a pool, bidding systems don't make sense to use.

2.3 Differentiation

What is being done differently in this project versus the myriad things done (or attempted) in experiments, field trials, competitions, and theoretical work with these areas of the Assignment Problem?

- This is the First Use of Assignment Problem technology in Science Olympiad for Event Assignment and Team Selection.
- The constraints used in Event Assignment include student availability and pre-set Event testing conflicts. Course capacity is set based on a uniform distribution of students across Events, not popularity of Events or the physical capacity limit of classrooms.
- The average Misery Index of the entire student pool is used to determine the quality or fitness of the Event Assignment. This is a *non-linear* objective function.

- Event Assignment is done entirely in the field, with actual students (and parents) entering their personal preferences based on their personal valuations for Events rather than in laboratory settings with artificially-induced subject valuations. In addition, the students must live with the outcome of their decisions for months, rather than merely the duration of a laboratory experiment.
- Changes and modifications to Event Assignment are actually implemented and affect the participants. This is very much unlike the studies on course assignment where analysis is done on the data, recommendations are made for improving the system and tested in the lab, but apparently the recommendations are rarely adopted by the institutions.
- Performance-based Team Selection is done using both *relative* and *absolute* performance measures.
- A tiered series of teams are formed from the student pool, where the performance of each team can actually be measured both in an absolute sense and relative to the other tiered teams. Hence, it can be determined if the team score is a reasonable proxy for the performance potential of the tiered teams.
- Student pairs on teams are intentionally paired with partners of equal ability to minimize the free-rider effect.
- If a student studied an Event, they should be allowed to compete in that Event. While a stronger team could be created by disallowing certain students from competing in one or more of their Events, it goes against the spirit of the SO competition to do so.
- This project weaves together Event Assignment, performance measurements, and Team Selection into a cohesive whole within an institutional structure rather than as standalone elements.

3 The Model

This dissertation focuses on two models within an institutional framework. The institutional framework was designed as part of this project to impose a formal management structure on, and consistent rules and procedures for, what was otherwise a chaotic morass. The two models work within the realm of SO. The first model involves assigning students to Events (courses[19]) based on the students' personal preferences. The second model involves forming competitive teams based on the performance measures of the students in their assigned Events. Both of these endeavors require assigning students to slots, where there is a cost associated with each possible assignment. To optimize the assignments, mathematical models of the tasks must be created. The two models each use different inputs and constraints to achieve their goals, as well as different sources for the costs associated with each possible assignment. The models each use different objective functions to measure the efficacy of an assignment.

SO is offered at some schools as part of their regular curriculum and at other schools as an extra-curricular activity. For those schools with SO as part of their regular curriculum, scheduling the teaching of Events is relatively easy since the Events are all studied during a normal elective period, when all students participating in SO are present in the class period. At these schools,

[19] In SO, courses are termed 'Events'.

Event teaching timeslots and student availability are not considerations for Event Assignment.[20] For Team Selection, students are more readily moved between teams and switched between their competitive Events, and test conflicts are soft constraints.[21] While the software and algorithms in this dissertation can be used to assist these schools, their environment is much simpler and less interesting than at schools where SO is offered as an extra-curricular activity.

At schools where SO is offered as an extra-curricular activity, Event Assignment for the student pool is a non-trivial exercise. Students at these schools can have limited schedule availability, students are not able to participate in 5 or more Events, and Event teaching timeslots are highly constrained. The student pool can be up to 10 times larger than the elective schools' student pools as well. Historically, it took between one and two weeks for parent volunteers at these schools to set up a schedule and assign students to Events—and the results of their efforts were often not pleasing to a majority of the participants. Similarly, putting together a strong team from a large, less-flexible student pool is exponentially more difficult than forming a team from a small, flexible student pool. SO as an extra-curricular activity is a much richer and challenging environment in which to work, and it is this environment that is the focus of this dissertation.

3.1 Misery Index

Before describing the model, it is important to define the measure that is used to evaluate the efficacy of the solutions from the Event Assignment. This measure is dubbed the "Misery Index" (MI). It calculates the relative 'misery' of a student based on that student's preference rankings for the Events to which they are assigned. This measure was chosen as it has proven to be understandable and considered 'fair' by the participants. The MI was also deemed acceptable as an aggregate efficacy measure during committee discussions. Note that MI is similar to one of the metrics used by Budish [6] to evaluate Course Assignment at the Harvard Business School (HBS).

Students are asked to arrange all 23 SO Events[22] in rank order of their personal preference from their most-favored Event to their least-favored Event. The system then assigns each of the Events an ordinal rank from 1 to 23, where the most-favored Event is assigned the ordinal rank of 1, and the least-favored Event is assigned the ordinal rank of 23. When the student is assigned to Events, the ordinal ranks associated with those Events that are assigned to the student are summed up to arrive at the total Misery Value (MV) for that student. A lower MV means the Events the student was assigned were closer to the top of their preference list, and conversely a higher MV means the student was assigned Events closer to the bottom of their preference list.

The Misery Index (MI) for each student is defined as the sum of their MV divided by the number of Events they are assigned[23]. For students with 3 Events, the minimum MI is 2.0[24], and the

[20] Essentially, all students are available in all timeslots and all Events are taught in all timeslots.

[21] Students have extra flexibility because they are each preparing for more than 5 Events (but will only compete in 3 or 4 Events), which makes it easier to move students between Events based on performance and work around any test conflicts they might have.

[22] For this thesis, it is assumed that a school will compete in all 23 SO Events. Some schools do not compete in all 23 Events, so the students would rank only those Events that the school decided to compete in, resulting in fewer than 23 Events being rank-ordered by the student. Whether all 23 Events or a subset are involved, the concepts discussed here are applied exactly the same way.

[23] In statistics, this is commonly known as the Average of a set of numbers.

[24] $1 + 2 + 3 = 6/3 = 2$, where a student is assigned their top 3 preferred Events

maximum MI is 22.0[25]. While most students in SO opt for 3 Events, there are many students who opt for 4, 2, or 1 Event, so those students' MIs occupy slightly different ranges and are also skewed compared to the MIs of 3-Event students. Equation (1) calculates the MI for a single student:

$$MI = \frac{\sum_{e=0}^{E} pref_{se} * x_{se}}{k_s} \qquad (1)$$

Where

s – a student

E – the number of Events

e – an Event

$pref_{se}$ – the preference rank for student s of Event e

x_{se} – binary indicator of whether or not student s has been assigned to Event e

k_s – the number of Events assigned student s

Students with differing numbers of Events make the MI an imperfect comparison value between students with different numbers of Events[26]. By the definition of the MI calculation, students with more Events will tend to have larger MI values than those students with fewer Events. It could be possible to compare students with different numbers of Events directly by modifying the calculation of the student with the greater number of Events to only include the same number of Events as the student with the smaller number of Events. However, comparing one student's MI to another student's MI is not a factor in this system. Rather, it is the MI of the whole pool of students in one solution versus the whole pool of students in another solution that is the important comparison.

So, when looking at the pool of students in different solutions, the *average* MI of one solution is compared to the *average* MI of another solution in order to determine which solution is preferred. A comparison of the total MVs of students in one solution is not compared to the total MVs in another solution.

3.1.1 Change to Misery Index

When this project started 7 years ago, there were just over 180 students applying for 150 slots in the SOP, where *every* student was requesting 3 or 4 Events. The MI as originally formulated worked well in that environment. As the student pool decreased in size and the students began requesting fewer than 3 Events, the MI skew started to become more apparent. Particularly in the last two years, the number of 1- and 2-Event students started approaching 1/3 of the student pool.

To account for the change in demographic, the definition of MI is changing on the live website for the 2021 SO season. The denominator of the MI will remain the same. The numerator will

[25] 21 + 22 + 23 = 66/3 = 22, where a student is assigned their bottom 3 preferred Events

[26] For example, 4 students who were assigned their most favored Events, where the 4 students were assigned 1, 2, 3, and 4 Events respectively would have Misery Indexes of **1** [1/1=1], **1.5** [(1+2)/2=1.5], **2** [(1+2+3)/3=2], and **2.5** [(1+2+3+4)/4=2.5] respectively.

change to be the MV minus the sequential sum of n for the number of Events that a student is assigned rather than just MV. The new MI calculation[27] is thus:

$$MI = \frac{\sum_{e=0}^{E} pref_{se} * x_{se} - \sum_{i=0}^{n-1} i}{k_s} \qquad (2)$$

Where,

s – a student

E – the number of Events

e – an Event

$pref_{se}$ – the preference rank for student s of Event e

x_{se} – binary indicator of whether or not student s has been assigned to Event e

k_s – the number of Events assigned student s

n – the number of Events assigned to a student

i – a counting index

3.1.2 Preference Intensity

While ordinal ranking of Event preferences does not record the relative intensity of preferences for Events, given the age of the students it would be difficult to explain the more nuanced concept of intensity to them, and to get them to understand things like assigning high intensity to two (or more) closely-ranked Events would make those choices near "perfect substitutes" for the optimization system. This might result in them getting a lower ordinal ranked Event in order to add more surplus to the overall system.

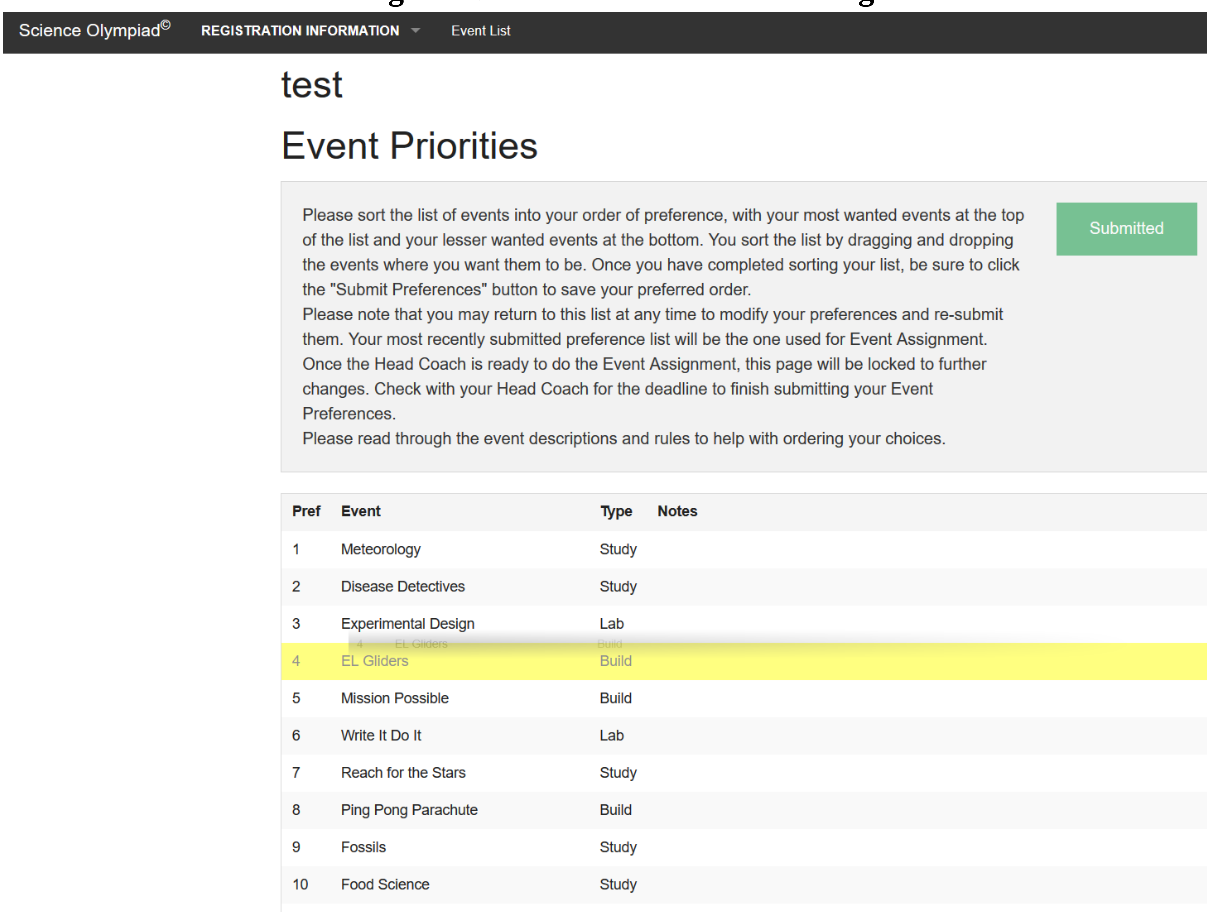
Looking at the experience of other experimenters with subjects using more complicated data entry options [29]: "Because of the complexity of the trading interface and of the intricacies of securities trading, recruiting was limited to those who had taken or were in the midst of taking courses related to finance." This is a quote about an experiment involving combinatorial packaging of a mere 3 goods and expressing intensity of interest in those goods or packages of goods, where as *few as* 18% and *no more than* 29% of the *trading orders* submitted by college students at Caltech who were enrolled in finance courses used *any* of the combinatorial features of the interface. Note the authors *did not* include any statistics on the *percentage of participants* in these experiments who entered combinatorial orders (which would have been insightful), only the percentage of order flow. This relatively low percentage of the more complicated order format shows that the experimental subject had to be a pretty *sophisticated* individual with a domain-specific educational background in order to have a chance of understanding the concepts covered in the experiment, let alone understand and operate the complicated user interface.

While Event Assignment is fertile ground for the use of combinatorics in specifying preferences for Events and groupings of Events, the participant pool of students ranges in age from 9 to 13 years old. For most or all of these students, their entire concept of an auction or the process of bidding is limited to hitting the "Buy It Now" button on eBay.com. It would be quite a stretch to

[27] Taking the example in the footnote above, the new calculation of Misery Indexes for the same 4 students are: **1** [(1-0)/1=1], **1** [((1+2)-(1+0))/2=1], **1** [((1+2+3)-(2+1+0))/3=1], and **1** [((1+2+3+4)-(3+2+1+0))/4=1] respectively. In the case where students get their least favored Events, the Misery Indexes are: **23** [(23-0)/1=23], **22** [((23+22)-(1+0))/2=22], **21** [((23+22+21)-(2+1+0))/3=21], and **20** [((23+22+21+20)-(3+2+1+0))/4=20].

imagine that all of these elementary and middle school students would be able to master the intricacies of not only ordering their preferences of Events but also signaling the intensity of those preferences across 23 items (more than 7 times as many items as a highly selective group of Caltech students were asked to handle) and do it in a meaningful way that accurately portrayed those preferences. There is no indication that the parents of these students would have fared any better in this task because they would have had to infer their child's preference intensities[28] and that also would not have yielded accurate portrayals. The complexity of the user interface would have increased significantly to accommodate an intensity measurement along with the lengthy accompanying explanatory text informing families about the ramifications and interactions of intensity values. And the user interface and instructions would increase exponentially more to add combinatorial features which few, if any, SO participants would likely use or be able to use correctly. Further, it would be an inherently *unfair* system if some of the participants were able to make good advantage of the more sophisticated features of such a system while most other participants would be placed at a marked *disadvantage* by not being able to employ those features.

Figure 1: Event Preference Ranking GUI



That all said, with 23 very different Events being ranked, there is an implied hierarchy of intensity: those at the top of the list are wanted much more than the Events at the bottom of the

[28] More than likely, many parents would have projected their own preferences and intensities onto their children in the ranking process.

list. That is, a student would be very happy with being assigned their 1^{st}, 2^{nd}, and 3^{rd} ranked Events, but rather unhappy with their 21^{st}, 22^{nd}, and 23^{rd}. For example, a student who does not want any Lab Events, in actual practice, ranks all Lab Events in the lowest ordinal spots, which very effectively expresses their dislike of those Events and minimizes their chance of being assigned one of those Events.

Figure 1: shows a simple drag-and-drop GUI where the students put their favored Events on the top of the pile and their least favorite on the bottom of the pile is *straightforward*, easy to explain, and an activity with which students in this age range are very familiar. The students understand the task and can accomplish the task easily and quickly—and they can easily understand the results.

3.2 Uniform Distribution of Students Across Events

As the number of participants in SO at the schools has declined[29] over the last 6 years, it has become more both more important and more difficult to distribute the students evenly across all Events.

Achieving a uniform distribution of students across Events is more important because teams are formed using students from every Event. If an Event has a small number of students enrolled in it, then those students are virtually guaranteed to be on a high-tiered team almost regardless of their performance. Similarly, if an Event has a large number of students enrolled, some of those relatively high performing students may be fit into a lower-tiered team, and some of those students may not be able to compete in that Event at all.

For example, say Event A has 16 students and Event B has 8 students, and there will be 6 teams formed. At most, 12 students from Event A will be able to compete in Event A[30], and at least 4 students will not be able to compete in Event A (whether they are on a team or not on a team). With Event B, up to 12 students could compete in Event B, but only 8 are available. Most likely, all 8 students in Event B will end up on teams, whether they are paired on 4 teams or spread out across all 6 teams. Simply because there are more students in Event A, there are bound to be a good number of high performing students[31] in Event A. And since there are fewer students in Event B, there is likely a lack of high performing students in Event B. When teams are formed, Event B severely constrains the possible sets of team members. If the distribution were even between Event A and Event B in this example, there would be 12 students in each Event. With 4 additional students in Event B (who, in this example, came from Event A), the average student ability in the Event will likely increase. When teams are formed, neither Event A nor Event B presents a binding constraint any worse than the other since there are an equal number of students to draw from either Event. In addition, most likely all of these students will be placed on a team and all of them will have the opportunity to compete in the Event for which they trained over the preceding 16 weeks.

---

[29] A new school was opened up, siphoning off roughly half the student body, dropping the number of students interested in SO from 180 to roughly 100, where the new school now has roughly 110 students interested in SO. The total number of students in the area has increased overall (also increasing the total number of students interested in SO), but now they are spread across two schools instead of being concentrated in one school.

[30] 6 teams with Event A * 2 students (max) per team for Event A = 12 students (max) competing in Event A.

[31] If Event A is a popular Event, such as Anatomy for aspiring student-doctors, then there is often an oversupply of high performing students in that Event. Whereas an unpopular Event often ends up with a disproportionate number of below-average students. It is not clear why this happens, as the students in the less popular Event most often have that less popular Event highly ranked in their Event preferences.

It is more difficult to achieve a uniform distribution because there are fewer students taking 3 and 4 Events, and more students taking 1 or 2 Events, resulting in fewer overall seats needing to be filled in each Event. This enlarges the impact of the 'remainder' problem: where there are empty seats in some Events after Event Assignment is complete. The empty seats may not be uniformly distributed across most Events, but could be concentrated in a few Events, which causes a few Events to under-enrolled compared to other Events.

As an extreme but illustrative example, say there is a pool of students who, combined, submit a total of 91 seats requests[32] for 10 Events. Assuming a uniform distribution of these students across all Events gives a need for 9 seats in each of the 10 Events[33]. Since it is desirable to have students paired up in all Events (or because it is desired to accommodate the 1 student seat-request remainder from the initial seat calculation), it is decided to make 10 seats available in all 10 Events: a total of 10 * 10 = 100 seats. This now leaves a surplus of 9 seats: 100 available seats – 91 seat requests = 9 surplus seats. Since this is an extreme example, say that 9 of the 10 Events are very popular, and 1 Event is exceedingly unpopular. So, Event Assignment in this case might fill 9 of the 10 Events to their capacity of 10 students, leaving one student (only) in the 10th Event. During Team Selection, this solo student would effectively be forced on to a high-tiered team as described in the previous example two paragraphs above.

There are a few possible solutions to this remainder problem for Event Assignment with regard to achieving a uniform distribution, presented here in decreasing order of viability:

1. Customize the Event seating capacities such that there will be no empty seats remaining. This can be done by using too few seats to meet the student demand[34], or through a little guess and check with the Event Assignment software to set Event capacities at or 1 below the capacity suggested by the SOTM software.
2. Accept the non-uniform distribution of students and recruit students to fill the empty seats.
3. Accept the non-uniform distribution of students as is and accept the subsequent negative impact on Team Selection.

The remainder problem wasn't an issue when there were more students (who were also requesting more Events on average 6 years ago versus now). This can be seen since there were 150 students requesting about 3.33 Events on average for 150 * 3.33 = 500 seats. This yields a class capacity of about 22 seats per Event (500 seats/ 23 Events = 21.7 seats/Event). Since at most 12 students from any Event can compete at the Regional competition, there was a 'surplus' of 10 students per Event. If one Event had 24 students and another had 20 students, the differential did not cause a binding constraint on team selection—certainly not anywhere near the problem that occurs when one Event has 8 students and another Event has 16 students.

3.3 Recommended Seating Capacities

The calculation for determining the recommended number of seats per Event is shown in equation (3):

[32] Sum the number of Events wanted by each student in the pool.
[33] 91 seat requests/10 Events = 9 seats per Event, with a remainder of 1 student seat request that can't be fulfilled.
[34] This is not a recommended solution, as the whole point of the SOP is to expose as many students as possible to STEM and provide them with the SO experience.

$$seatsPerEvent = \frac{\sum_{s=0}^{N} maxE_s}{num2studentEvents + 1.5 * num3studentEvents} \qquad (3)$$.

Where,

N – the number of students

s – a student

$maxE_s$ – the maximum number of Events desired by student s

num2studentEvents – the number of Events which allow a maximum of 2 students to compete

num3studentEvents – the number of Events which allow a maximum of 3 students to compete

seatsPerEvent is the recommended number of seats per 2-student Events, and 1.5**seatsPerEvent* yields the recommended number of seats for all 3-student Events. These two values are displayed on the Event List web page of the SOTM as a recommendation to the head coach on setting the maximum capacity for the Events that are being offered by the SOP in the current season.

3.4 Event Assignment

SO Event Assignment is an SPAP (Student-Project Allocation Problem) where multiple students can be assigned to a project (in this case, a project is a competitive Event).

The goals set out for Event Assignment when this project was started in order of importance are:

1. Give Students the Events they want. Students who get their favored Events are more likely to be enthusiastic learners.
2. Be as Fair as possible. Participants need to feel that they are being treated equitably with everyone else.
3. Adhere to SO Competition Rules. Students are not assigned to Events in which they cannot compete.
4. Uniform distribution of Students in Events. Students are distributed across all Events uniformly so that there is not a subsequent restriction on team selection.

The primary focus is thus on the benefit of the students rather than on the potential competitiveness of a team in a competition. Thus, there is *not* a goal to 'rig' Event Assignment to achieve a pre-determined Team Selection result; there is *not* a goal to favor 'top' students; and there is *no requirement* that students re-take Events in the current year that they performed well in during a prior year. These goals (and excluded goals) were designed to avoid enabling parental behavior which, in prior years, led to confrontations and unhappy students.

Thus, our objective here is to assign students to their more favored Events subject to a series of constraints around the number of Events desired by the student, student availability, schedule of Events, Event testing and teaching conflicts, team composition considerations, and incentive considerations. Except for team composition and incentive considerations, all of these constraints are hard constraints with no room for flexibility.

The objective function minimizes the aggregate MI over all students in the pool so that the overall pool is as well off as it can be. The aggregate MI of the pool is calculated by summing the MIs of the individual students.

Some constraints enforce the student-set limitations and the testing and teaching conflicts. Other constraints assist with student distribution amongst the Events to make team selection more flexible. The Event capacities and limits can be calculated to create recommended values that will tend to make conditions for team selection more favorable as shown in section 3.3. However, the actual capacities used are set by the SO head coach for a school.

The incentive considerations pander to those conditions believed to induce adults and students to adopt behavior that is beneficial to the SO program and promotes greater flexibility in team selection. For example, as an incentive for getting parents to coach an Event, the child of an adult who volunteers to coach will receive preferential treatment during Event Assignment. These optional incentive considerations are discussed in more detail in section 3.4.3. The incentive considerations are optional and are used at the discretion of the SO head coach for a school.

It is also important to note that Event Assignment is performed in a multi-round format rather than as a straight-up IP optimization. This is because of the non-linear objective function, and the incentive considerations which require sequence-specific operations. It is not clear that a specialized non-linear integer program can be designed to solve this genre of problem with desirable results.

3.4.1 Event Assignment Objective Function

The objective function for Event Assignment is:

$$minimize\colon \sum_{s=0}^{N}\left(\left(\sum_{e=0}^{E} pref_{se} * x_{se}\right)/k_s\right) \qquad (4) \quad .$$

Where

N – the number of students

s – a student

E – the number of Events

e – an Event

$pref_{se}$ – the preference rank for student s of Event e

x_{se} – binary indicator of whether or not student s has been assigned to Event e

k_s – the number of Events assigned student s

3.4.1.1 Non-linearity of the Objective Function

A unique feature of the objective function is the k_s denominator. Since k_s is a variable[35], it makes the objective function non-linear. The goal of the objective function is to minimize the MI of the entire student pool, not to minimize the MV of the student pool. Therefore the MI of each student must be calculated for use in the objective function.

Minimizing the MV will yield different results from minimizing the MI. Minimizing MV can and will yield multiple solutions with the same MV, which means that, mathematically, the solutions are ‘equivalent’ even though they can be quite different qualitatively. Minimizing MI

[35] k_s varies for each student based on the number of Events that they are assigned in any particular solution.

tends to yield a solution where the MIs of the students in the pool tend to be more closely clustered around the objective function's MI rather than being more widely distributed across a wider range of student MIs.

As an extreme example, assume a pool of 3 students is taking three, two, and one Events, respectively. Assume there are a total of 6 Events being ranked in preference order for each student and that all students have the same order of preferences. Assume each Event has a capacity of one student. The following tables show some of the possible solutions with the MV and MI objective values calculated for each solution.

In the tables, Students are listed across the row. The Assigned Event Preferences columns show the preference rank of the Events that a student was assigned. The Student MI and Student MV columns show the MI and MV for each student based on their assigned Events. The Total MI and Total MV columns show the value calculated for the student pool for an objective function based on MI and MV.

In the three cases shown in Table 1:, Table 2:, and Table 3:, the Total MV is the same. To select the best solution from a group of tied solutions, either a secondary criterion would need to be introduced to break the tie or the preferred solution would be the first solution found with the minimum Total MV. There is nothing wrong with having a secondary criterion for breaking ties, it's just cleaner and more preferable to have a single criteria.

Table 1: Maximum Misery Index Solution

Student	Assigned Event Preferences			Event Count	Student MI	Student MV	Total MI	Total MV
1	1	2	3	3	2	6		
2	4	5		2	4.5	9	12.5 (4.2)	21
3	6			1	6	6		

The Total MI is different in each case. The preferred solution is then the solution with the lowest Total MI shown in Table 2:. This is labeled as an Unbounded solution as this solution cannot be reached with the Event Assignment algorithm implementation[36], but it is a solution that could be reached through an IP implementation of Event Assignment. The RSD Event Assignment implementation could arrive at the solution shown in Table 3:.

Table 2: Minimum Misery Index Unbounded

Student	Assigned Event Preferences			Event Count	Student MI	Student MV	Total MI	Total MV
1	4	5	6	3	5	15		
2	2	3		2	2.5	5	8.5 (2.8)	21
3	1			1	1	1		

Also note that in the real world, the scenario of multiple students all choosing the same preference ranking for all of their Events is astronomically unlikely. So, as different Events are assigned to different students in the various solutions, the MV and MI of students will vary from solution to solution, which also means there will be some variance in Total MV and Total MI

[36] As discussed in section 3.4.6, the Event Assignment is implemented as an RSD, not an IP.

between solutions. Since MV is an integer calculation, there are likely to be multiple ties with the solutions; however since MI is a decimal calculation, it is less like that there will be exact ties—especially with larger student pools.

Table 3: Minimum Misery Index By Round

Student	Assigned Event Preferences			Event Count	Student MI	Student MV	Total MI	Total MV
1	3	5	6	3	4.7	14	8.7 (2.9)	21
2	2	4		2	3	6		
3	1			1	1	1		

In addition, in the real world, since there are multiple seats available in each Event, 90+% of students are typically assigned their 1st choice Event—actually it has always occurred that all students have been assigned their 1st available Event choice. It is typically in the second round of Event assignment that students start diverging as to which preference ranked Event they are assigned next. Hence, this example is extreme in that only one student can be assigned their 1st choice Event, which exaggerates the relative difference in MV and MI between students compared to what students experience in the real-world implementation.

3.4.1.2 Objective Function Value Context

As a raw number, the objective function value can vary greatly from school to school and from year to year. This is because the value depends on the total number of students, the number of Events the students want, and the number of Events that they are assigned (which depends on the Event Assignment constraints). As such, the raw objective function value has no context and can't be compared with other SOP's results: it's just a number that needs to be minimized.

It is easy to bring context to the objective function value, which is done by simply dividing the objective value by the number of students, *N*, that are being run through the assignment. This yields a proxy value for the average MI of the entire pool of students in the Event Assignment. This number is now directly comparable with other schools' Event Assignment results as well as being a useful value for providing context of individual students' Misery Index values from the Event Assignment for comparing how an individual student fared compared to the average of the overall pool.

For example, the objective function might report a value of 645, but it is not immediately apparent whether this is a relatively good result or a relatively poor result. However, if there are 150 students in the pool, this yields an average MI of 645/150 = 4.3, which is pretty good. However, if there were only 70 students in the pool, the average MI is 645/70 = 9.2, which is pretty bad.[37] In this example, while the reported objective value for the two student pools is the same, the quality of the two results is very, very different.

Thus, the raw objective value doesn't provide context as to what a good result is versus a poor result, but the average MI calculated from the raw objective value does provide significant insight. With a poor result, modifications can be made to the configuration to work toward a

[37] From section 3.1, for a student with 3 Events, the best possible MI is 2, and the worst possible is 22. From experience, the best average MI's from a well-arranged Event Assignment is in the low 3's to low 4's. A poorly configured Event Assignment will result in average MI's ranging from the low 6's to the 11's.

better result; while a good result means the Event Assignment can be considered done and the head coach can move on to other tasks.

3.4.2 Event Assignment Constraints

The Event Assignment is subject to these conflict constraints, student-imposed constraints, and distribution constraints:

Table 4: Event Assignment Conflict & Student-imposed Constraints

Hard Constraints	Description		
$\sum_{e=0}^{E} x_{se} \leq maxE_s$	A student may not be assigned to more Events than they request.	(5)	.
$\sum_{e=0}^{E} x_{se} * b_e \leq maxB_s$	A student may not be assigned to more build Events than they request, where Build Events ⊂ All Events	(6)	.
$x_{se} - \sum_{a=0}^{A} v_{sa} * t_{ea} \leq 0$	A student may not be assigned an Event that is taught outside of their schedule availability.	(7)	.
$\sum_{e=0}^{E} x_{se} * g_{be} \leq 1$	A student may not be assigned an Event that has a test conflict with their assigned Events (over each testing bloc *b*).	(8)	.
$\sum_{e=0}^{E} x_{se} * g_{te} \leq 1$	A student may not be assigned an Event that has a teaching conflict with their assigned Events (over each teaching bloc *t*).	(9)	.
$\sum_{s=0}^{N} x_{se} \leq c_e$	No more students can be assigned to an Event than the capacity for that Event's classroom.	(10)	.
$\sum_{s=0}^{N} x_{se} * f_s \leq L_e$	No more than L_e students per Event *e* shall be students with 2 or fewer requested Events.[38]	(11)	.

Where

$maxE_s$ – the maximum number of Events desired by student *s*

$maxB_s$ – the maximum number of build Events desired by student *s*

b_e – a binary constant indicating whether an Event *e* is classified as a build Event or not

v_{sa} – a binary constant indicating whether or not student *s* is available in timeslot *a* or not

t_{ea} – a binary constant indicating whether Event *e* is taught during timeslot *a* or not

g_{be} – a binary constant indicating whether Event *e* is in the testing bloc *b* or not

g_{te} – a binary constant indicating whether Event *e* is in the teaching conflict bloc *t* or not

c_e – is the maximum capacity of students for an Event *e*

[38] As discussed in section 3.2, team formation can be greatly affected if the distribution of students across Events is not uniform.

f_s – is a flag indicating whether or not a student has requested two or fewer Events

L_e – is the limit of flagged students in an Event e

3.4.3 Event Assignment Optional Incentive Considerations

Optional incentive preference and penalty considerations are shown in Table 5:. The preferences move the students with those characteristics to the front of the assignment list in one or more assignment rounds. The penalty causes students with that characteristic to be skipped for an assignment round.

Table 5: Event Assignment Optional Incentive Considerations

	Optional Incentive Preferences	Description
i.	Coach's Kid	Children of coaches are assigned Events before all other students in the first round of assignment.
ii.	4-Event	Students who request 4 or more Events are assigned Events before all other students in the second round of assignments.
iii.	2-Event Penalty	Students who requested 2 or fewer Events are skipped in the second round of assignment.
iv.	Grade Level	Students in higher grades chose first over students in lower grades. (not available in SOTM)
v.	2-Event Limit	Limit the number of students who requested 2 or fewer Events who are assigned to each Event.

With the exception of the 2-Event Limit, these constraints which were put in place to provide incentives and considerations for the targeted participants do not provide the hoped-for benefits to the SOP, but rather result in negative consequences for the overall institution.

The objective in Event Assignment is to assign students to Events that are as high on their preference list as possible. The theory being that a student will be a more enthusiastic learner and participant if they study what they want rather than being placed in an Event they are uninterested in.

Thus, the goal of Event Assignment is *not* to make Team Selection easier by slotting students into Events in such a way as to make the students more replaceable as components of a team. In fact, a 'simple' method for making students easily-assigned perfect replacements for each other would be to group Events together in flights[39] of 3 or 4 Events at a time. Any student on any team could be swapped for another student on another team with the same flight of Events. Students could then rank these flights in order of their preference and then be assigned a flight of Events that fit their schedule availability. While this method works for schools where SO is part of the curriculum, this does not work for schools where SO is an extra-curricular activity. The

[39] The flights of Events would obviously be grouped together such that the set of Events in each flight did not have testing conflicts.

students' availability would severely restrict their opportunity to be assigned to the flights, where the less available a student was, the more likely they couldn't be assigned any flight.

While it might be possible to collect all the students' availability and then construct flights around the pool of availability, this then would necessitate a multi-stage registration task. It would take a week or more just to collect student availability[40] which is critical to design a set of imperfectly compatible[41] flight groups. It would then take another week or more to collect the students' customized flight preferences (in place of the students' Event preferences). This process would consume a lot of calendar time, and can cause logistic problems if some families miss one or more of the stage of deadlines not to mention the administrative headaches that then go with accommodating those students who missed deadlines.

It is simply not practical and less outcome-efficient to set up an Event Assignment mechanism built around Team Selection. Thus, an Event Assignment that is almost completely blind to Team Selection is implemented.

3.4.4 Inputs: Preferences, Availability, Schedule, and Conflicts

The inputs into the model come from students, parents, coaches, the school, and the National, State, and Regional SO organizations. Some of these inputs interact with other inputs, and some of the inputs are independent.

Event Preference Rankings, maximum Event counts, and student availability originate from students and their parents. Each student has their schedule of timeslots when they are available to participate in SO, and each student has their personal preferences for which Events are their more favored Events and which are their least favored. Preferences and schedule availability may be influenced to greater or lesser degrees by parents and/or friends of the student.

The Schedule is a relatively fixed set of input based on the availability of parent coaches to teach a particular Event. A schedule is set up consisting of a series of 1-hour to 3-hour long blocks dividing up the after-school hours, evening and weekend hours into timeslots for scheduling the 23 Events for their teaching period. Each Event is assigned to one timeslot, with some timeslots having multiple Events assigned to the timeslot. Parents are asked to provide up to 3 possible timeslots in the block schedule when they are available to teach their Event. Typically the parents provide only 1 timeslot as being available. If it is necessary to move an Event to a timeslot other than the one specified by the parent (due to classroom availability or too many Events being scheduled in the same timeslot), that timeslot move must be individually negotiated with the parent(s) concerned. Hence, it is not possible to use a timetabling algorithm as there are no scheduling options available without manual intervention through individual negotiation.

[40] Student Registration is held open for SO for 7 days, during which time 95+% of the students who expressed interest in SO complete all steps of Registration. One year, an attempt was made to shorten Registration down to 4 days by announcing a week in advance and on the Registration page that the window was 4 days; however, only 70% of the students completed Registration in the shorter window. Registration had to be re-opened for 4 more days to allow another 28% of the interested students to complete Registration. Based on this, there is little doubt that a multi-stage Registration process would span no fewer than 2 weeks.

[41] Students would be forced to take specific combinations of Events, where some Events in those flights might be of very low interest to the student. This would most likely result in much lower MI scores for students than they have from Event Assignment done without flights.

Figure 2: Example Grid of Event Testing and Teaching Conflicts

Test Conflicts				
Block 1	**Block 2**	**Block 3**	**Block 4**	**Block 5**
TH WIDI ED M	MM MA SS	H CB DD	DP E ANAT RM PP	FF HP RS O

Teaching Conflicts				
1	WS	T		
2	DD	CB	H	BB
3	RC	ED		

Event Testing Conflicts arise from groups of Events that are tested in the same testing bloc at competitions. Since a student cannot be in two places at the same time to take tests, a student may only be assigned one Event from those Events listed within a testing bloc. Most of the time, the testing blocs set by the National SO organization are also observed by the State and Regional organizations; however, sometimes State and Regional will deviate from National's blocs, and then the superset of Regional, State, and National testing bloc conflicts need to be observed. An example of test conflicts is shown in the left-most table of Figure 2:. The testing blocs are numbered and shown by color-keyed columns. Each Event in a bloc is listed by an acronym. Note that only 19 of 23 Events are listed in the table. The 4 Events not listed are tested throughout the competition day rather than during any specific bloc. These are known as *self-scheduled* Events and are displayed in **black** font on the right side of Figure 2:). The 4 self-schedule Events do not have hard test conflicts with the other 19 Events.

Event Teaching Conflicts arise when two or more Events that are not in the same Test Bloc are taught during the same schedule timeslot. An example of teaching conflicts is shown in the right-most table of Figure 2:. The teaching conflict groups are listed in the numbered rows and each row represents a different timeslot that contains teaching conflicts.

3.4.5 Schedule Visible or Hidden

There were two possible interactive modes available when students register for SO to indicate their preferences for Events and their schedule Availability:

- Event Schedule Visible – the timetable displays the names of the Events in the timeslots in which they were being offered
- Event Schedule Hidden – the timetable does not display any Event names on the Schedule

For these two modes for the Event schedule display, there are positive and negative trade-offs:

1. *Visible* – families have the opportunity to rearrange their family schedule to free up the timeslots where their favored Events are scheduled; however, some families might (severely) limit their schedule availability based on which timeslots their favored Events are offered. For example, a family with 8 timeslots available might artificially restrict their availability only to the 3 timeslots in which their student's favorite 3 Events are being offered. In this mode, the Event schedule must be finalized before opening student registration.

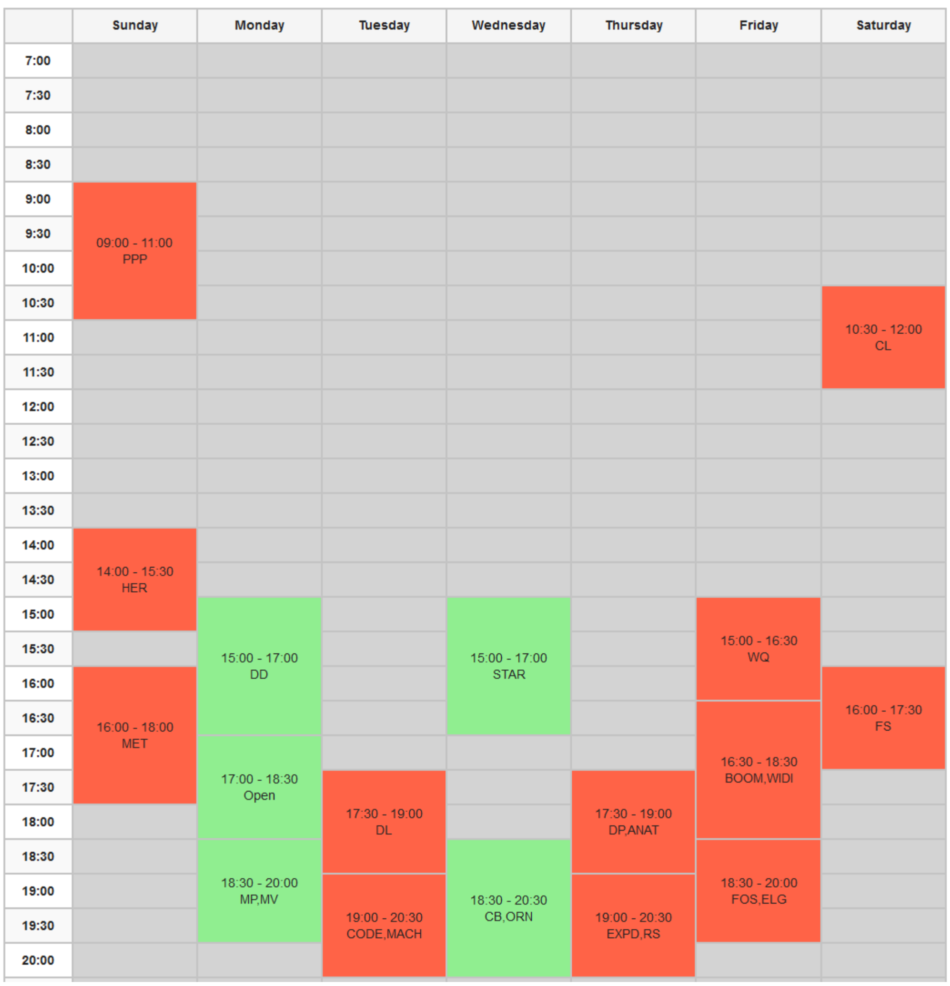
2. *Hidden* –families will tend to mark more timeslots as being available when the Events are hidden than when they are visible; however, they cannot rearrange other family activities around the SO schedule in order to make their student available for a highly desired Event. For example, if a family knew when the Events were offered, the family might move a music lesson from one day to another so that their student could be enrolled in Astronomy during the timeslot that their music lessons normally occurred. In this mode, the Event schedule need not be finalized before opening student registration.[42]

In either mode, parents and students are informed that the best way to get assigned Events as high as possible on your list of preferences was to reveal truthfully:

1. Your preferences for Events
2. Your actual schedule availability

They are also advised that the more schedule availability that a student has, the more likely the student would get the maximum number of Events that they signed up for, especially because of testing and teaching conflicts between Events.

Figure 3: Schedule Availability GUI



[42] It is necessary that at least as many timeslots be present in the schedule as will be required to support the final Event schedule.

The decision to display or hide the Event schedule is left up to the Head Coach.

Figure 3: shows the GUI where students enter their schedule availability by clicking on the timeslots where they are available for SO activities. This is an example where the Event names are displayed in their teaching timeslots; in the scenario where the Event schedule is hidden from the participants, the same schedule timeslots are shown, but the Event names are removed. Green timeslots indicate that the student is available and red timeslots indicate that the student is unavailable.

Here again, it would be possible to have students register their ordinal preference and even their intensity of preference for the various timeslots. Similar to ranking Event preferences, the level of sophistication required of the student participants and the complexity of the user interface to accommodate that activity would be higher than can be reasonably expected from students in their age group.

A simple on/off toggle GUI where the students click 'green' for the timeslots they can attend class and 'red' for those they cannot attend is *straightforward*, easy to explain, and an activity with which students in this age range are very familiar. The students understand the task and can accomplish the task easily and quickly—and they can easily understand the results and see that they were not assigned any Events outside of their availability.

3.4.6 Random Serial Dictator

A serial dictator is a mechanism for distributing items amongst a group of agents. It is 'serial' in that one agent is dealt with at a time. It is 'dictatorial' in the sense that a dictator determines the order in which each agent chooses an item from the remaining set of items. A serial dictator is a 'truthful' (or strategy-proof) mechanism since, when an agent is choosing, the agent clearly will be best off truthfully selecting its most-favored item from the remaining items rather than concealing information by selecting a less-favored item.[43] The order in which the agents act can be determined in innumerable different ways, as may be appropriate for the domain area and/or the rules that are set for the serial dictator to operate under.

A Random Serial Dictator (RSD) uses a random draw to determine the order of the agents. Since the order of agents is randomly set from a uniform draw, an RSD is considered 'fair' in that the agents are treated as equals and, thus, each agent has the same chance of appearing in any slot of the sequence as every other agent in the group.[44]

The algorithm for performing the Event Assignment was selected to be a Random Serial Dictator (RSD) rather than a non-linear integer optimization for several reasons:

- An RSD is easy to explain to unsophisticated users and is perceived to be "fair"
- An RSD is quick and easy to program, easy to extend, and easy to debug
- An RSD is one of the more efficient methods of allocation [16]
- An RSD solves quickly in a known amount of time: $T(n) = O(n)$, so it can be run multiple times from different starting points and the solutions compared to see which solution is more optimal

[43] As discussed in section 2.1.4, multi-round SDs are potentially open to gaming. The reward is being potentially better off than a truthful reveal, against the risk of being potentially *much* worse off than a truthful reveal. The consequences of gaming in this project are discussed further in sections and subsections of 4.1.1.

[44] https://en.wikipedia.org/wiki/Random_serial_dictatorship.

- An RSD can be modified to be pseudo-random (instead of strictly random) to take into account the incentive considerations
- An RSD does not require any special purpose software (such as IP software)
- It is not clear that a specialized non-linear integer program can be designed to solve this genre of problem with guaranteed optimal results.

The algorithm for the RSD is straightforward. Students are assigned Events in a series of rounds, with the number of rounds equal to the maximum number of Events wanted by the pool of students (typically, this is 4 rounds, although it has been 5 rounds on two occasions). In each round, the students are assigned a single Event (if possible according to the constraints). Once all rounds are finished, the assignment is complete and the results can be reviewed.

Within each round, the following steps occur in the RSD:

1. The students are sorted into precedence groups[45].
2. Each precedence group is sorted into a randomized order[46].
3. The highest precedence group remaining is processed one student at a time in its randomized order:
    a. Select the student's highest-ranked Event remaining in their preference list
    b. Check if there are seats still available in the Event
    c. Check if the student is available in the Event's timeslot
    d. Check if the student has a test conflict with an Event they've already been assigned
    e. Check if the student has a teaching conflict with an Event they've already been assigned
    f. If this is a build Event, check if the student has reached their maximum number of build Events
    g. If all of the checks pass, then assign the student to this Event and go to the next student.
    h. If any check fails, mark this Event as not allowed for this student, then go to Step (a) above and repeat with the next most favored Event in the student's preference list
    i. If the end of the student's preference list has been reached, then it is not possible to assign this student another Event, and they are complete
4. Once a precedence group is done with the round, then go to Step 3 and repeat with the next precedence group

[45] There is a feature in the RSD software which allows students to be divvied up into *N* precedence groups, which, when carried to the limit, allows the system administrator to specify an exact order in which students will be processed in each round. Precedence groups are processed one group at a time in order from highest to lowest precedence. In implementation on the SOTM website, this precedence grouping is limited to the incentive considerations only, which means there are at most 2 precedence groups in any one round. These typically are:

- Round 1 - coaches' kids and everyone else
- Round 2 – everyone else and 2-Event students (who are skipped)
- Round 3 – 4-Event students and everyone else
- Round 4 – 4-Event students only (everyone else already has all their Events assigned)

If Round 5 is needed, only those students requesting 5 Events need to be processed.

[46] Note that the randomized ordering of students is different every round so that the first students processed in one round are not necessarily the first students processed in other rounds.

The vast majority of students are assigned the number of Events that they want, and generally get the Events which are the higher ones on their preference lists. Due to any of several factors (including the constraints, the varying popularity of Events from year to year, and individual student input), not all students may get their desired number of Events. However, the RSD will *never* assign a student to any Event which they cannot attend or in which they would not be able to compete.

The RSD is fed an initial random seed. This random seed is used to initialize the randomization used by the RSD. Therefore, the results from the RSD are perfectly repeatable when the RSD is re-run with the same seed and same input data, but a different feasible solution is generated when a different random seed is used.

3.4.7 Optimization

While an RSD is an efficient method of allocation, there is no guarantee that the results are optimal or even near the optimal solution. In addition, due to the nature of ordering and processing the students sequentially, most different random orderings of students will potentially end with significantly different assignment results for the various students. The quality of these different solutions is viewed subjectively by the students involved, who clearly would favor those solutions that treated them best regardless of the results for the overall pool or other individual students.

Having declared the MI of the student pool as the objective measure of quality of each solution, the MI is used to compare the solutions. Thus the solution with the lowest MI is the best result. However, since the RSD is not a non-linear/integer programming algorithm, it's very unlikely and certainly not mathematically convincing that any single run of the RSD is optimal.

To search for an optimal result, the RSD can be set to run through multiple iterations, each iteration using a different starting random seed. The MI generated by each iteration can be compared, and, after *N* iterations, the iteration with the lowest MI is closest to optimal. The RSD implements this by taking the initial random seed and incrementing it by 1 each iteration. Due to the nature of the random function in the library, a difference of 1 in the random seed produces a very different ordering of students between iterations. While this method does not exhaust the solution space, practical experience shows that rarely is there significant improvement of the MI after 2,000 iterations—even if the iteration count is increased to 100,000. For practical purposes, a 'very near' optimal allocation solution can be achieved in 2,000 iterations, which usually takes less than 30 seconds to run. 30 seconds is a reasonable waiting time for a head coach to wait in front of their web browser for a result. It is also a short enough time that the head coach can play around with those assignment parameters that are under their control in order to examine alternative high-quality solutions.

Table 6: provides a look at the improvement in the MI vs. running more random iterations on the solutions for Event Assignment. The Iteration Cutoff is the number of iterations that were run to get the baseline MI. Iteration Limit is the total number of iterations that were run to see if a better MI could be achieved. The Average Number of Improvement Incidents is the average number of iterations in which a better MI was achieved during the additional iterations, where the average was computed over the total number of trial runs. Average Improvement Percentage is the average improvement in MI between the baseline value and the final MI value achieved with the additional iterations. Maximum Improvement Percentage is the maximum improvement in the MI over the baseline value across all trial runs on that row. Trial Runs are the total

number of samples that were run for each configuration row. From the table, it can be seen that the improvement in MI stays below a maximum improvement of 2%, with an average improvement of less than 1%. Given that each iteration takes between 0.013 and 0.035 seconds to run, a 2,000 iteration solution takes between 26 and 70 seconds to complete. Increasing to 10,000 iterations would increase the run-time by a factor of 5 to between 2 and 6 minutes for a run; and increasing to 100,000 iterations increases the runtime by another factor of 10 for 20 to 60 minutes per run. A 1 or 2% increase (at most) in MI doesn't warrant the additional 4 minutes to 1 hour of waiting time that would be imposed on adult volunteers who are planning on doing test runs and what-if scenarios before settling on their final set of parameters. Not to mention delaying the report of the assignment results to the students and coaches, and delaying getting the students started in their sessions.

Table 6: Misery Index Improvement with Increase in EA Iterations

Iteration Cutoff	Iteration Limit	Average Number of Improvement Incidents	Average Improvement Percentage	Maximum Improvement Percentage	Trials Run
2,000	10,000	1.45	0.68%	1.6%	20
5,000	20,000	1.17	0.40%	0.7%	12
10,000	50,000	1.60	0.52%	1.4%	20
10,000	100,000	2.42	0.80%	1.8%	40

In a typical Event Assignment, not all students get all the Events that they want[47]. The difference in solutions with a MI change of less than 1% is reflected by the swapping around a handful of the students who were short Events with students who were not short Events. Similarly, a handful of students swap an Event to get a more preferred Event with other students who end up with a less-preferred Event. The delta change for a few individual students can be fairly large[48], but most are relatively small differences. Thus, while a longer run-time for the solution yields a more optimal solution, it is not substantially different from any other solution within 1% of its value with regard to the overall student pool.

The MI of the student pool is an imperfect measure of solution quality because the measure favors solutions where not all students are assigned as many Events as they want. For example, solution **A** where every student is assigned their top 3 Events has a MI of 2, whereas solution **B** where every student is assigned only their top Event (even though the students all want 3 Events) has a MI of 1. By straight comparison of the Misery Indices, solution **B** is superior even though the students are all short 2 Events. Clearly, this is not true to the spirit of the Event assignment goals.

It would be possible to add an MI weighting of some value to those students who are short Events, but the selection of that value would be arbitrary and certainly would not reflect the fact that some students might prefer being short an Event than being assigned an Event below their,

[47] However, *most* students do get assigned the number of Events they are requesting.

[48] For example, one student's Misery Index improved by 4.5 points by dropping an Event that was low on their priority list (and not replacing it with a different Event), while another student's Misery Index worsened by 6.5 points by adding the lowest ranked Event on their preference list. However, most of the changes are less than a 2 point swing in Misery Index for any of the affected students.

say, 10^{th} preference choice. In forcing such an arbitrary weight onto the MI, the algorithm would then favor solutions that avoided penalties. This can only be accomplished by favoring solutions where highly constrained students essentially ‘pick first’ and flexible students pick from the leftovers. This results in dropping some of the flexible students significantly down their preference lists in order to provide a small number of inflexible students with Events (which also might be very low on their preference lists) in order to avoid any MI penalty, whether implicit or explicit. Thus, the overall pool’s MI could be lowered by up to the penalty amounts in order to avoid the penalty. This is not an assignment strategy that would make the student pool, as a whole, happier.

With the current Event Assignment implementation, those students who end up short on Events are those students with limited availability and thus constrain the system rather than add flexibility. It is usually not considered a good thing to reduce group surplus in order to increase the surplus to a highly constrained individual when the addition of that individual does not add to the overall surplus of the system.[30] In fact, such a mechanism as this *drains* the surplus by up to the amount of the penalties. Fortunately, the RSD algorithm implementation does not hunt for solutions where more students are short on Event counts. The sequential round nature of the algorithm strives to fulfill all of the student pool’s Event requests before calculating the MI. The algorithm will not produce a degenerate solution[49] while minimizing the MI.

To that end, the optimization choices have expanded with the addition of five criteria from which to choose. The choices of optimization of the Event assignment are:

1. Minimize the MI
2. Equalize Distribution with the minimum MI – as a first objective, this optimization measures the number of empty seats in each Event and prefers any solution iteration where the delta between the Events with the fewest and most empty seats is minimized. If solutions have the same delta value, then the solution with the lowest MI is considered better. This objective works best for distributing students as evenly as possible across Events and is useful where there tends to be a population bias in preferred Events and/or schedule availability. Note that setting the Event capacities to their suggested values greatly assists in achieving this goal of uniform distribution.[50]
3. Maximize Student Fill with the minimum MI – as a first objective this optimization measures the total shortage of Events assigned to students. The lowest total shortage is considered best. If solutions have the same total shortage, the solution with the lowest MI is considered better.
4. Maximize Class Fill with the minimum MI – as a first objective this optimization measures the total number of unfilled seats across all Events. The lowest total is considered best. If solutions have the same total, the solution with the lowest MI is considered better.
5. Minimize Unpartnered Students with the minimum MI – as a first objective this optimization measures the number of Events with odd numbers of students enrolled. The lowest total of odd enrollments is considered best. Note that since there is no way to predict or control the number of students who drop out of the SO program or drop individual Events, this

[49] For example, no student being assigned to any Event, which would yield a Misery Index for the student pool of zero.
[50] See section 3.3 on recommending Event capacities.

optimization path cannot guarantee that fewer students will be competing unpaired after Team Selection.

While objectives 3 and 4 seem to be striving for the same performance quality (unfilled seats/Event requests), they can generate different solutions from each other.

3.5 Schedule Optimization

Occasionally, there is some flexibility in the timetable of which Events are offered in which timeslot. This can be optimized relatively simply and quickly by moving Events between timeslots and rerunning the assignment. Some rudimentary analysis on the available student data can be performed to get an idea of which courses are most (and least) popular and which timeslots have the most (and least) student availability[51]. By moving around some Events to different timeslots, it is possible to minimize students being short Events and to minimize the overall MI.

The more flexibility in the Event schedule (timetable), the less trivial this task is.

3.6 Team Selection Optimization

SO Team Selection is a resource allocation/assignment problem where students from a pool are assigned to teams based on student performance in order to always make the strongest potential team from the remaining student pool.

The goals set out for Team Selection when this project was started were (and still are):

1. Adhere to the Competition Rules. To participate in the SO competitions, a team must be legal.
2. Create Teams with the Strongest Potential based on recorded Student Performance. To advance to the next level of competition, a team must be made with the strongest combination of students that cover all Events. Hence the combinatorial nature of this task.
3. Create a Fully Staffed Team. To advance to the next level of competition, a team must be able to compete in and be competitive in *every* Event.

In addition, there are goals specific to selecting teams for the Regional Competition:

1. Create a Tiered Series of Teams.
2. Pair Students of Equal Ability. This tends to minimize the Free Rider issues.
3. Students are allowed to Compete in the Events for which they studied. If a student studied an Event, they should be allowed to compete in that Event. While a stronger team could be created by disallowing certain students from competing in one or more of their Events, it goes against the spirit of the SO competition to do so.

At all levels of competition, the goal is to provide the school with its strongest potential team so that it has the best opportunity/highest probability for advancing to the next level of competition. This must be done in an objective manner by distancing human bias from the process as much as possible, hence the reliance on an algorithm to select teams based on reported student performance.

[51] For example, see Table 13:, Table 14:, and Figure 9:, Figure 10:, and Figure 11:.

Specifically, the objective is to form a tiered-series of teams from a given student pool (without replacement) such that the team with the highest potential performance[52] is created from the remaining student pool. The constraints on team selection are built around students and their potential combinations of Events, team composition rules of the National SO, and the incentive considerations that the head coaches choose to implement.

The objective function is to maximize the Team Score, which is the sum of the weighted performance of students in their Events for all students assigned to the team who are able to compete in the Events.[53]. The performance scores of the students are weighted to account for each student's relative contribution toward the performance of the student pairings and to take full advantage of the rules of SO.[54] The performance scores of students in their Events are assumed to be a reasonable proxy for that student's future performance in a competition. The objective function is:

$$maximize\colon \sum_{e=0}^{E}\sum_{s=0}^{N}\sum_{w=0}^{W_e} h_{ew} * \; skill_{se} * x_{sew} \qquad (12) \quad .$$

Where

E – the number of Events

e – an Event

N – the number of students

s – a student

W_e – the maximum number of students allowed to compete in an Event

w – a 'seat' in an Event

h_{ew} – the weighting of this student's skill in this Event

$skill_{se}$ – the reported total score/performance of this student in this Event

x_{sew} – binary indicator of whether or not student s has been assigned to Event e with weight w for this team

3.6.1 Team Selection Constraints

The team selection algorithm is subject to the constraints in Table 7:.

[52] An alternate strategy would be to select all teams such that they had approximately equal potential. While this would be an equitable distribution of students, it is not a competitive distribution of students with regard to teams from other schools. As an example, let's say 4 teams were to be made from a pool of students totaling 2000 points. In a set of equal potential teams, each team would total about 500 pts (2000 pts/4 teams = ~500 pts/team). However, a tiered set of teams might have team scores of 800, 600, 400, and 200. Assuming the team score is a good proxy for team performance, the 800-pt team would handily beat all 4 500-pt teams—and it is likely that the 600-pt team would best all 4 500-pt teams as well. The 800-pt team would likely advance to the next level of competition, whereas it is unlikely that any of the 500-pt teams would advance.

[53] Recall that there is a maximum number of students on a team who are allowed to compete in each Event. Since there can be more students on a team trained in an Event than the maximum allowed to compete, the Team Score will only include the highest performance scores up to the maximum number of students allowed in each Event.

[54] Event performance weighting is discussed in more detail in sections 3.6.3 and 4.2.6.

Table 7: Team Selection Constraints

Hard Constraints	Description	
$\sum_{s=0}^{N}\sum_{w=0}^{W_e} x_{sew} \le W_e$	Limit the number of students assigned to an Event to the maximum allowed to compete in that Event.	(13) .
$\sum_{w=0}^{W_e} x_{sew} \le 1$	Limit each student to be in only one of the weighted categories per Event.	(14) .
$\sum_{s=0}^{N} x_{sew} \le 1$	Limit to one student per weighted category per Event.	(15) .
$\sum_{e=0}^{E}\sum_{s=0}^{N}\sum_{w=0}^{W_e} x_{sew} \le 15$	Limit the number of students on a team.	(16) .

Note that the constraints related to testing and teaching conflicts and limiting the maximum number of Events that a student can compete in were handled by the Event Assignment and thus can be ignored by the Team Selection IP (TSIP).[55]

3.6.2 Team Selection Parameters

Unlike Event Assignment, where the head coach can change Event capacity and other parameters to limit enrollment of students, there are no such parameters for Team Selection for the head coach to modify. For example, the head coach cannot limit the number of 2-Event students on teams, nor can the head coach change the maximum number of students on a team that are allowed to compete in an Event[56].

3.6.3 Team Selection Optional Considerations

For team selection, a few optional considerations can be incorporated into the TSIP. Most of these are primarily designed to maximize student participation in competitions, whether by increasing the number of students involved in the teams or by increasing the number of Events

[55] A head coach can make manual Event Assignment changes for students using the SOTM software. Manual changes are not checked for conflicts, so conceivably a student could go into Team Selection with a conflicted set of Events and potentially be placed on a team where that conflict would be an issue. The head coach is warned of this possibility and the consequences by both the SOTM GUI and the SOTM Head Coach Guide documentation. It is up to the head coach to honor the conflicts—or not. To date, all manual Event changes have observed the conflict constraints.

[56] One year, a head coach wanted one Event to have only one student per team competing in that Event, with the stated rationale being that only 6 students were in the Event and it was desired to have that Event represented on all 6 teams at the Regional competition. The head coach wanted to keep this change for the State Team selection, even though the 6 students were sufficient to fill the Event with 2 students for State. This was a clever head coach: it turns out that the only way to get his kid on the top Regional Team was this change. Similarly for State, the only way to get his kid on the State team was with this change. The head coach had had an employee write a team selection algorithm, and then had spent an entire day for both Regional and State playing with parameters to get his desired result. As discussed in section 3.2, an Event with low enrollment can constrain the solution and result in a different team roster than a less constrained solution. Setting the maximum student count in that one Event effectively doubled the enrollment for that Event and relaxed the constraints from that Event. Since this occurrence, head coaches are no longer able to adjust the maximum number of students competing in an Event.

that the student pool as a whole competes in. Other considerations exist for incentive or performance reasons.

The optional incentive and performance considerations are:

Table 8: Team Selection Optional Incentive and Performance Considerations

	Optional Preferences	Description
i.	Regional or State Student Performance Weightings	Use the Regional or State set of weightings (h_{ew}) for the objective function.
ii.	Writers and Doers	For the Event "Write It, Do It", treat the two student roles as if they were separate Events.
iii.	Coach's Kid	Force a coach's kid onto a team.

i. The Regional and State weights are different values used in the objective function to tilt the objective function into achieving different results.

The Regional weights are set such that all Events and students are weighted equally, which tends to pair students of equal ability in Events within a team. Here the weights are all set to 0.5 for all $\sum_{w=0}^{W_e} h_{ew}$ for all Events. This results in all two-student Events having a combined weight of 1.0, and all three-student Events having a combined weight of 1.5. When there are more students than spaces available in teams, this weighting does a good job of pairing students of equal ability with regard to their ordinal rankings within the Grade book for their Event. As the number of students remaining in the pool approaches the number of spaces available on a team, the limited number of pairing possibilities makes equal ability pairing less likely to occur. As an example, with 100 students vying for 15 spots on a team, the pairing of equal ability is fairly likely to occur. However, with 20 students vying for 15 spots, the equal pairing depends entirely on who is left in the pool. So, when forming 6 teams from a pool of 100 students, the first 3 to 4 teams have reasonably good equal pairing, while the last 2 to 3 teams formed are less likely to have as much equal pairing. Note also, that since this is a maximization problem, the objective function still selects the highest-scoring team based on the student pool and their Event performance even if that means having unequal students in one or more or all Events.

The State weights are set such that higher-performing students are preferred over equally performing students. This is accomplished through making the weights within an Event unequal, but still maintaining equal total weight between Events (based on the number of students in the Event). Two-student Events still have a combined weight of 1.0 and three-student Events still total to 1.5: however, within the Event, the weights are unequal. For example, build Events might have weightings of 0.8 and 0.2, study Events with 0.7 and 0.3, and lab Events with 0.6 and 0.4, and 3-student Events might have weights of 0.6, 0.5, and 0.4.

How effective are these weightings in making differences in the actual team selection? It depends on the size of the student pool and how skewed the weightings are. The larger the pool and/or the more skewed the State Weightings are from the balanced Regional Weightings, the more likely teams selected using the two different weightings will be different. In general, a larger pool size has more effect with a smaller skew, but a small pool generally requires a larger skewing. With a pool size of around 45 students, the skewing needs to be pretty extreme in order to cause different team rosters to be generated with the two weightings.

ii. The Writers and Doers refers to the "Write It, Do It" (WIDI) Event, which involves one student writing directions about how to assemble an artwork and a second student following the directions from the first student to recreate that artwork sight unseen. Some WIDI coaches consider these skills not to be interchangeable. They strongly requested that any team made be composed of one Writer and one Doer for the WIDI Event. This option treats the WIDI Event as two separate sub-Events with a single student participant in each, each with a total weight of 0.5 (which keeps WIDI in total equally weighted with all the other 2-student Events). The students involved in WIDI are also identified by the WIDI coach as Writers or Doers and have their performance scores allocated to their respective sub-Event. The Team Selection then proceeds with a (likely) guarantee[57] of having one Writer and one Doer on each team.

iii. Coach's Kid for Team Selection is present for the same reason it is present in Event Assignment: it is an incentive for coaches to continue coaching after teams are selected. This option causes a coach's kid to have a very large bonus[58] added to the objective function upon their selection to a team. This is done only when the final team is being formed.

The optional participation considerations are:

Table 9: Team Selection Optional Participation Considerations

	Optional Participation Preferences	Description
iv.	Full Team	Force a team to use all 15 students
v.	Drop Minimum	Force a team not to drop a student below N Events
vi.	Solo (Unpartnered) Competitors	Minimize the number of solo competitors

iv. The Full Team option makes the team size requirement to be exactly 15 team members, where equation (16) has its inequality sign changed from $\leq$ to =. This forces the IP to assign exactly 15 students to the team. The goal of this option is to maximize the number of students who participate in competitions. This option generally will cause the score of a team to decrease when this is a binding constraint (that is, without this option, the optimal team would have had fewer than 15 team members *and* a higher score).[59] This option is used only in the case where: all teams have been selected, not all teams have 15 team members, and there are still students in the pool who were not assigned to a team. In this case, the team selection would be re-run where this option would be exercised

[57] Obviously, if there are not as many Writers or Doers as there are spots available on teams, there may be one or more teams without their complementary WIDI partner. Hence the need for the "Solo (Unpartnered) Competitors" option in Table 9:. Note that all other Events are also subject to having unfilled partners for the same reasons, which also points to the need for the "Solo Competitors" option.

[58] A bonus of 1000 points is added to each of the coach's kid's Event performance scores. Since performance scores always range between 0 and 100, the kid is virtually guaranteed to be placed on the team. The exception being where there are more than 15 coaches' kids vying for the final team or there are 3 or more coaches' kids with overlapping Events.

[59] See section 3.6.5.5 for a detailed example of why a team that is forced to be full-up can have a *lower* Team Score than a team that is not full.

starting with the first team to have fewer than 15 students through the final team to be selected.

v. The Drop Minimum option adds a new constraint to the team selection such that if a student is assigned to a team, then that student will compete in all of their Events on that team. This constraint is:

$$\sum_{e=0}^{E}\sum_{w=0}^{W_e} x_{sew} - u_s * c_s \geq 0 \qquad (17)$$

Where u_s is whether or not the student is assigned to the team and c_s is the minimum of the number of Events the student is enrolled in and the minimum number of Events a student is allowed to compete in.

In the standard team selection run, a student can be assigned to compete in anywhere from 1 to the maximum number of Events they are enrolled in. This can result in a situation where a student has studied for many weeks in four Events, but is selected for a team where that student only competes in one of their four Events. As this can be very demotivating for a student—particularly for Regional Competitions—the team selection can be re-run with this option enabled so that a student does not drop below a minimum threshold of Events in which they compete. It is possible to set the threshold so that no student drops any Event, thus allowing each student to compete in all of their Events. This constraint can cause teams to have lower total scores and be less competitive than the optimal team selected without this constraint being imposed.

Note also that this constraint can be implemented such that only 4-Event students drop at most one Event. While the goal of this constraint is admirable (that a student should be allowed to compete in the Events for which they have studied for weeks), the decreased competitiveness of the team is not desirable. The consequence of imposing this constraint is that there are some Events where there are more students on a team than allowed to compete in that Event. In the unconstrained optimal team selection, the lowest scoring student(s) in those over-populated Events would act as alternates for the higher scoring students in that Event. However, with this option enabled for 4-Event students, it is not unusual for a 4-Event student to have to 'drop' over-populated Event A in favor of a second student with fewer than four Events who is also in Event A but who has a lower score in that Event than the 4-Event student. This situation can lead to problems between the parents of the affected students, as well as between the affected Event coach and the head coach.

vi. The Solo Competitors option minimizes the number of Events where students on a team must compete without one (or more) of their partners. Since all Events are designed to be competed in with pairs (or triples) of students, solo competitors generally compete at a disadvantage. Thus, if the team selection results in solo competitors being on a team, this option can be enabled and the team selection re-run starting with the first tiered team to have solo competitors through the final team to be selected.

This option is implemented through the modification of the objective function where a bonus is added if an Event is fully staffed with its maximum complement of students. The following elements are added to the objective function:

$$+\sum_{e=0}^{E}\sum_{w=0}^{W_e} bonus_e * bonusPts_e \qquad (18)$$.

Where

bonus$_e$ – is a binary variable that only is true if all student spots in an Event are filled.

bonusPts$_e$ – is the number of bonus points allocated to each full Event

And these constraints are added:

Hard Constraints	Description	
$0 \le \sum_{s=0}^{N}\sum_{w=0}^{W_e} x_{sew} - bonus_e * W_e \le W_e$	If an Event is not fully staffed, no bonus points are counted in the objective function.	(19) .

A final note on the team selection model is that some schools have students prepare in up to 8 Events which allows considerably greater flexibility in optimal team formation. For example, if there two student pools of the same size and equivalent ability, where one pool has students with 8 Events each and the other pool has students with 4 Events each. It's easy to see how the 8-Event-per-stduent pool has many more combinations and permutations of students that can form a legal team than does the 4-Event-per-student pool. The 8-Event pool is able to form a team at least as good as or better than the 4-Event pool's best team. The 4-Event pool's best team might equal the 8-Event pool's best team, but 4-Event team will never be better than 8-Event team.

3.6.4 Inputs: Student Performance, Weights, Considerations, and Team Constraints

The inputs into the Team Selection model come from Event coaches (student performance in class), competitions, head coaches, and the National SO organization. There are rubrics that determine the total score for each student in each of their Events.

Student performance is measured by entering scores into a Grade book. There is a set of Grade book scores for each Event for all the students in that Event. The coach for the Event is primarily responsible for entering scores for their students based on homework, quizzes, and/or tests. Each Event coach enters the values for the rubric for their Event to weight the import of homework, quizzes, and tests, and the Event coach determines the point values of the various assignments and the points earned by each student for those assignments. In addition, the Event coach is responsible for entering citizenship and attendance scores for each student in their Event. An example Event rubric is shown in the left-most table in Figure 4:.

The Grade book also contains the scores earned by students in the competitions in which they participate. The performance results at competitions are converted into scores and entered into the Grade book by the head coach. The head coach is also responsible for setting the Master Rubric which is used to combine the Event Grade book scores[60] and the competition scores to compute a total score for each student in all of their Events. A student's total score for an Event is the parameter used in the objective function for Team Selection. An example Master Rubric is shown in the right-most table in Figure 4:.

[60] This includes the assignment, citizenship, and attendance scores entered by the Event coaches.

Figure 4: Example Event and Master Rubrics

Rubric Locked for Event Coaches.

	Weight %	Drop Lowest	
Homework	5.00	☑	
Quiz	45.00	☑	
Test	50.00	☐	Save

Rubric

☐ Lock Rubric

Add Competition

Element	Date	Regional Relative Weight	Regional % Weight	State Relative Weight	State % Weight	Droppable	Special Config	Notes
Attendance		5	5.0%	5	1.6%		☑	Average over all of a Student's events
Citizenship		5	5.0%	5	1.6%		☑	Average over all of a Student's events
Gradebook		30	30.0%	30	9.4%		☑	Normalize Gradebook for Composite Score
Mini Comp test	2019-11-23	30	30.0%	30	9.4%	☐	Remove	
Mini Comp 2	2019-11-24	30	30.0%	30	9.4%	☐	Remove	
CCA Invitational	2020-1-11	0	0.0%	60	18.8%	☐	Remove	
Regional		0	0.0%	160	50.0%			
State		0	0.0%	0	0.0%			
National		0	0.0%	0	0.0%			
Totals		100	100%	320	100%			

0 Maximum Number of Competitions to Drop

Reset Save

The Event weightings used are solely based on whether the Regional Teams are being selected or the State Team is being selected. The head coach is not allowed to pick and choose or otherwise modify the Event weightings.

The head coaches may select the optional considerations they deem appropriate.

The team constraints are set by National SO. The fundamental constraints typically won't change from year to year; however, the number of 3-student Events can change from year to year, which changes the number of student-Events needed on a team and in turn impacts the number of 4-Event students needed on a team.

3.6.5 Example Team Formation

The easiest way to understand Team Formation is to do a series of simple examples of the concept. In each example, students are displayed along the rows and Events are shown in columns. For these examples, each student will be enrolled in two or fewer Events. A student is shown as being enrolled in an Event if there is an X or a number in the row/column cell where the student's row intersects with the Event's column. In Table (I) of Figure 5:, Student B is enrolled in Events 2 and 4, and Student F is enrolled in Events 1 and 2. A team is then composed of no more than 2 students in each Event and no more than 4 total students on the team.

For this series of examples, the first step is to determine which combination of students can form legal full teams. The second step is to determine which legal full team scores the highest (optimum). The third step looks at partial team possibilities. The fourth step is to examine what can happen when students are not uniformly distributed across all Events. These four steps simply correspond to solving the assignment problem with variously relaxed constraints.

3.6.5.1 Event Coverage

In Table (I) in Figure 5:, there are 6 students (rows) and 4 Events (columns). Each student is enrolled in exactly two Events and each Event has 3 students enrolled. These enrollments are indicated by the 'X's in Table (I). The team formation rules require that no more than two students may compete in an Event and no more than 4 students can be on a team.

Figure 5: Team Formation: Event Coverage

Two Letters in Every Numbered Category

Uniform Distribution

	1	2	3	4
A			X	X
B		X		X
C	X			X
D		X	X	
E	X		X	
F	X	X		

(I)

A + B + E + F

	1	2	3	4
A			X	X
B		X		X
C	X			X
D		X	X	
E	X		X	
F	X	X		

(II)

A + C + D + F

	1	2	3	4
A			X	X
B		X		X
C	X			X
D		X	X	
E	X		X	
F	X	X		

(III)

B + C + D + E

	1	2	3	4
A			X	X
B		X		X
C	X			X
D		X	X	
E	X		X	
F	X	X		

(IV)

Looking at the possible combinations of students and Events that satisfy these requirements, it can be seen that there are at most 3 combinations of students, which are shown in Figure 5:'s Tables (II) with students A, B, E, and F, (III) with students A, C, D, and F, and (IV) with students B, C, D, and E (selected students have their letters highlighted in yellow/tan). Any other combination of students will violate one or more of the team formation rules: either too many students on the team or too many students in an Event. While it is possible to have fewer than 4 students on a team or fewer than two students in an Event, the next example will show that those partial teams are not necessarily optimal.

3.6.5.2 Scoring

In Table (I) in Figure 6:, scores have now replaced the enrollment X's for the students. In each of the Events, the students are scored relative to each other with the scores having been 'normalized' across all Events: a score of '3' (green) is the highest score in the Event, and a '1' (red) is the lowest score. The scores are entered from each student on the team for each possible team configuration in Tables (II), (III), and (IV). The column scores for each Event are added up and totaled at the bottom of the Tables and the total score for the team is summed up from the column scores and displayed above each team Table. The team from Table (II) is the optimal team as it has the highest team score of the three possible teams.

Figure 6: Team Formation: With Scoring

(I)

	1	2	3	4
A			3	1
B		1		3
C	1			2
D		3	1	
E	2		2	
F	3	2		

(II) Score 17

	1	2	3	4
A			3	1
B		1		3
C				
D				
E	2		2	
F	3	2		
Col Totals	5	3	5	4

(III) Score 16

	1	2	3	4
A			3	1
B				
C	1			2
D		3	1	
E				
F	3	2		
Col Totals	4	5	4	3

(IV) Score 15

	1	2	3	4
A				
B		1		3
C	1			2
D		3	1	
E	2		2	
F				
Col Totals	3	4	3	5

3.6.5.3 Highest Ranked Students

As alternative teams, two teams composed of the students with the highest scores are selected, where one team consists of those four students who scored a '3' in one of their Events, and the second team consists of four of the five highest scoring students not used in the Figure 6: teams nor the team of students scoring '3's. These teams are shown in Figure 7:. This is choosing teams by 'rank' order rather than optimization.

With the team in Table (I) of Figure 7:, there are a few conditions to note. First, the team score is 16, which is lower than the score of 17 for the optimal team in Figure 6:. Second, Event 1 only has a single student competing in it—although it is the highest scoring student in that Event it is still a handicap during the competition[61]. Third, there are three students trained in Event 2, but only 2 students may compete, so the lowest scoring student, B, has their score set to 0 as they are not competing in the Event and thus do not add to the scoring potential of the team.

Figure 7: Team Formation: Ranked Teams

(I) Score 16

	1	2	3	4
A			3	1
B		0		3
C				
D		3	1	
E				
F	3	2		
Col Totals	3	5	4	4

(II) Score 16

	1	2	3	4
A			3	1
B				
C				
D		3	0	
E	2		2	
F	3	2		
Col Totals	5	5	5	1

With the team in Table (II) of Figure 7:, there are similar conditions to note: the team score is 16 and lower than the optimal team; Event 4 has a single student competing in it *and it is the lowest performing student in the Event*; and Event 3 has three competitors, so student D is dropped from the Event. For this team, Event 4 is a larger handicap than is Event 2 in the first alternate team

[61] Student B could be moved into Event 1 and trained to assist in the Event.

since the student is the lowest performing student, and even if student D were added to Event 4, D is not trained and likely would not greatly increase the team's performance in Event 4.

There certainly could be situations where a ranked team or a partial team might approach or even equal the potential performance score of an optimal full team. However, the handicap from the partial Event fills would need to be weighed against full Event participation, where some Events are more amenable to late-entry competitors and others are not.

3.6.5.4 Non-uniform Distribution

For the non-uniform distribution case, the students are slightly re-arranged such that there is not an even distribution of students across Events, although the students still all were enrolled in 2 Events. The student/Event distribution from Table (I) in Figure 6: is modified by moving student A from Event 3 into Event 1. Since there are now 4 students in Event 1, the scores are modified so that Students A and C have the same score of 2, and student E is modified to have a score of 1. This keeps the scoring 'normalized' between the Events rather than expanding the scoring in Event 1 to range from 1 to 4 or 0 to 3, which would then over- or under-weight Event 1 relative to the other Events. The resulting distribution of students shows four students in Event 1, two students in Event 3, and three students in Events 2 & 4. These modifications are highlighted in the cells with orange numbers in Table (I) of Figure 8:.

Figure 8: Team Formation: Non-uniform Event Distribution

	1	2	3	4
A	2			1
B		1		3
C	2			2
D		3	1	
E	1		3	
F	3	2		

(I)

Score	16			
	1	2	3	4
A				
B		0		3
C				
D		3	1	
E	1		3	
F	3	2		
Col Totals	4	5	4	3

(II)

Going through the same selection process as in Figure 6: and Figure 7:, the optimal team is shown in Table (II) of Figure 8:. Similar to the Ranked Team selection, there are a few conditions to note: the team score is not as high as the original uniform student distribution's optimal team, Event 4 has only one student participating, and Event 2 has three students where student B is dropped from the Event. Unlike the Ranked Team selection, this is caused by constraints of the data rather than the methodology: Event 3 restricts the flexibility of the system and Event 1 has options that are unusable because of those restrictions. As a whole, team selection with a non-uniform distribution is significantly limited because of the imbalances. The constraints are worse as the student pool gets smaller, and they are less of a factor with larger student pools. This is very well illustrated by a Regional team selection which starts with unbalanced Event rosters, where the imbalance is not noticeably constraining on the first team or two, but as the pool shrinks, partial teams and solo Events become more common with each successive team created.

3.6.5.5 Forced Full

This example demonstrates that a team that is forced to be full can have a *lower* Team Score than a team which is not forced to be full. Assume a student pool with three students, where two students have three Events each, and the third student has two Events. The 3-Event students are the high scoring students and the third student scores lower. The team can be composed of a maximum of three students and can compete in up to three Events with a maximum of two students allowed to compete per Event. The student pool and their Events are shown in Table 10:. The students are listed in rows, with their respective performance scores in each of their Events listed in the numbered (Event) columns.

Table 10: Forced Full Student Pool

Student	1	2	3
A	10	10	10
B	9	9	9
C	8	8	

Team Members	Score
A+B	57
A+C	46
B+C	43
A+B+C	56

The optimal team is composed students A and B (only), as their combination yields the maximum Team Score. Any other combination of students results in a lower total Team Score. This includes using the maximum number of allowed team members (three students), meaning students A, B, and C. This is because for student C to be on the team, student C *must* compete in at least one Event, which means that either student A or B must *not* compete in whichever Event(s) that student C would be slotted to compete in. Since student C has a lower performance score than both A and B in all Events, student C is dominated by both A and B. Thus, there is no combination of students such that student C improves the Team Score.

If there were a combination of students from the pool that yielded a higher Team Score *and* resulted in a full complement of team members, the TSIP would have found it and presented it as the optimal solution. If the optimal Team Score is found and it results in a team roster that is less than the maximum team size, then that, in fact, will be the optimal solution: a team with fewer than the maximum number of allowed members.[62]

Similarly, an optimal team could be composed such that there were no student competitors at all in one or more Events. This is seen from live data from 2015 in Table 43:, where the unconstrained TSIP yielded an optimal Team Score with one unfilled Event, three partial Events, and a 54-point (8%) higher Team Score than the team formed using the Full Team option (which required it to fill all Events).

3.7 Crowdsource

The idea of using crowdsourcing to complement team formation was proposed by a parent at one of the very early meetings in which ideas were discussed on how to go about designing the Team

[62] This has happened in actual practice. This occurred when there was a student pool with an unusually large portion of 4-Event students who were also high performers. A typical SO team is composed of 13 3-Event students and 2 4-Event students. By including 3 additional 4-Event students on the team, then one 3-Event student could be eliminated. This would leave all the Events fully staffed, but the team would have 14 members instead of 15. Since the TSIP finds the optimal solution, that means that there was ***no*** *valid* solution using 15 students that *exceeded* the Team Score of the 14-student team.

Selection methodology, particularly in relation to selecting the State Team[63]. When this endeavor started in earnest in the summer of 2014, crowdsourcing was still relatively uncommon and primarily used in niche areas[64], so it was a novel idea to apply it to a performance-based team selection.

The idea was to bring the students and parents (stakeholders) in on the fun by getting them to understand and appreciate the complexity and difficulty of the team selection problem. Hopefully, this would get buy-in to the final result and, in small part, see whether any stakeholders could find good solutions. Humans are not particularly well equipped to find a *good* initial solution to this type of combinatorial problem, but they appear to be rather good at finding incremental improvements to an initially seeded solution.

The model here was to present the exact same student performance data and working constraints as are implemented in the Team Selection algorithm, seed an initial solution and then see what solutions people were able to generate.

The guaranteed payoff for the stakeholders was that if you selected a team that scored higher than the team selected by the algorithm, then your team would be the State Team.

To make it accessible and easy to use by the stakeholders, a web page was created and loaded with the same data fed to the Team Selection algorithm, complete with a display showing the objective function value generated by the human and the ability to pick and choose and modify team members within the constraints of team selection. The display is a table with Events listed by name in rows and students listed by an ID number in columns[65], with each cell in the table containing the total score of a student (column) in that Event (row). Clicking on the cells in the table would toggle a student in or out of an Event (likewise, toggling the student on or off the team), with the team score being updated on every click, and illegal combinations being disallowed. A complete description of the crowdsource tool web page is in Appendix F.

This crowdsource page allowed any stakeholder to pit their ideas for team selection methodologies against the algorithm. A few enterprising students and parents even scraped the web page for the raw data and equations and built spreadsheets or wrote programs to try to beat the algorithm!

3.8 Data Sources

Live data was collected for this project from students and parents at 6 schools between 2015 and 2020 (6 SO seasons of data). The schools were a collection of middle, elementary, and high schools. Not all schools participated in all features of the SOTM software, and some schools joined into the program over the duration of the project, so not all data is available from all schools for the entire period.

The sources of the data collected are shown in Table 11:.

Data from the 2020 SO season were collected from most of the schools listed here. The data is complete and is used in this paper for the Regional competition data analysis and State Team

[63] Selecting the members of the State Team was (and is) always the most contentious aspect of SO amongst the parents.

[64] One infamous example being the “Policy Analysis Market” (PAM) promoted by Robin Hansen and Net Exchange **Error! Reference source not found.**.

[65] This allowed the students (and their parents) to identify themselves based on their Event mix, but only make educated guesses about who the other students were.

Selection. However, the State performance data is not included due to COVID-19. The SO State and National tournaments were cancelled due to health concerns, so the final data collection for State performance was incomplete.

Table 11: Data Source Schools and Participation

School	Year	#	EA	TS	CS
CVMS	2015	7	Y	Y	Y
	2016	6	Y	Y	Y
	2017	6	Y	Y	Y
	2018	6p	Y	Y	
	2019	6p	Y	Y	Y/optimal
	2020	6p	Y	Y	Y
PTMS	2017	6p	Y	Y	
	2018	6p	Y	Y	
	2019	5p	Y	Y	
	2020	6p	Y	Y	
OAES[66]	2016	2	Tm	S	
	2017	3p	Tm	S	
	2018	3	Tm	S	
	2019	4p	Tm		
	2020	4p	Tm		
CCA[67]	2016	5p	Tm		
	2017	5p	Tm		
SCES	2018	2	Tm		
	2019	3	Tm		
	2020	3	Tm		
BHS	2019	3p	Tm		

Key	Description
EA	Event Assignment
TS	Team Selection
CS	Crowdsource Tool
Y (EA)	Assigned Events to Pool
Tm (EA)	Pre-selected Regional Teams and assigned Events within a team
Y (TS)	Selected both Regional and State Teams with algorithm
S (TS)	Only selected State Team with algorithm
Y (CS)	Used CS with State Team selection
Y/optimal	Optimal Team posted
(blank)	Did not use this feature
# p	# teams, p: some partial teams

3.9 Institutional Structure

Over the approximately 10 years that it was operating as an organization, CVMS SOP had evolved to be dysfunctional. There was no consistency from year to year. There was no institutional memory. There was a high and growing level of discord amongst the adult leaders and volunteers. And there were rocky and volatile relationships with the umbrella PTSA organization as well as the Regional SO. The problem resided mostly with a few of the chain of leaders of the SOP, who appeared to be more interested in using their position to get their child on the State team rather than trying to run the SOP in fair, performance-oriented manner.

To address the dysfunctional nature of the SOP and save the program from being cancelled by the school principal, a group of about 18 concerned parents worked together over the course of 5 months to establish an institution with reasonably clear rules and procedures that would be able to persist from year to year and operate transparently in all matters regarding the SOP.

The key features of the new CVMS SOP institution are:

[66] In 2019, OAES sent their highest scoring team to State, so they did not change teams after Regional.

[67] For CCA, Team 1 was selected and students were assigned Events by hand by the CCA SO student committee. Teams 2 through 6 were randomly staffed with students by the committee and then the EA program was used to assign Events to the students on those teams. Based on the results of EA, transfers of students between teams were suggested and most of the suggestions were accepted by the student committee, resulting in improved Misery Indexes for the students on those 5 teams. The State team was always selected by hand by the student committee.

- The Coordinators (aka head coaches) are elected by a voting body of current coaches rather than appointed by the prior year's Coordinators. The group of coaches are familiar with the workings of SO and returning coaches will be working with the new Coordinators, so the coaches have a vested interest in selecting reasonable Coordinators.
- Because the Coordinators are elected, they can also be voted out of office. Previously, it was not possible to remove a (rogue) Coordinator from their appointed office.
- Candidacy for Coordinator is open to any adult. All parents with a child currently in the SOP, as well as incoming 7th grade parents with a child in SO, are solicited to be candidates.
- The Coordinators are overseen by and report to a Steering Committee. The Steering Committee act as advisors and vote on significant issues and decisions for the SOP.
- The Steering Committee reserves a seat for a faculty or staff member appointed by the Principal from CVMS.
- Written Rules & Procedures are established, announced, and made available to all participants at the beginning of the school year. The rules and procedures were made as clear as possible to avoid as much ambiguity or last-minute interpretation as feasible. The rules and procedures documents are listed in Appendix B SOTM Rules & Procedures.
- Changes to the Rules & Procedures must be approved by the coaching voting body and the Steering Committee members.
- The focus was redirected back to the students' welfare. Participation was increased, students were not locked out because of lack of prior SO experience, and teams were selected based on actual student performance in the current year, not based on performance from the prior year and/or personal acquaintances.
- Efforts are made to eliminate the bias and favoritism that had been readily apparent in prior years.
    - *All* students are assigned to Events by a computer algorithm based on their stated preferences and availability.
    - Teams are picked by a computer algorithm based on their current year performance and constrained by the Regional, State, and National SO team requirements.
    - The coach voting body votes to approve the final *process* used for selecting the State Team.
- Most significantly, the Head Coordinator's ruling is final, and this is coordinated with and backed up by the CVMS Principal. If the members of the SOP strongly disagree with the Head Coordinator's decision, the coach voting body can vote to remove the Head Coordinator.

In many ways, the institution mimics a corporate governance structure, with a CEO (the Head Coach), a Board (the Steering Committee), and published rules and procedures. While the Head Coach and Assistant Head Coach run the day-to-day administrative activities, the Steering Committee exercises oversight on those activities and decisions as well as serving as a backchannel communication path with other families not on the Steering Committee. The Event coaches also have a roll in governing through voting on the leadership and procedures. In addition, students are encouraged to self-advocate in any disputes as parents are not allowed to file grievances on behalf of their students[68].

Lastly, the Head Coach is strongly encouraged to set all the Overall Rubric and Team Selection weights that effect Team Selection (whether Regional or State or both) as early as practical so that all stakeholders know the parameters when the season starts. Event Coaches are also asked to set their grade book rubrics by a deadline[69]. These parameters should be locked down long before any one has any knowledge of how student performance is shaping up, which then makes it difficult for a parent to complain that things were done unfairly.

All of these points make for an open, transparent institution. If these rules, procedures, and guidelines are followed, it becomes difficult for a Head Coach to manipulate the system to their own or another's personal advantage.

Since the initial implementation of this seminal institutional structure at CVMS, substantially similar institutions using SOTM have been adopted at PTMS, OAES, and SCES. A handful of other middle schools in the San Diego area that do not use the SOTM but have become familiar with the CVMS SOP structure through meetings and conversations during the last 6 years have adopted similar institutional frameworks.

4 Analysis

For purposes of this paper, the focus will be primarily on the SOP at CVMS as that is the school where data, details, and experience were initially gleaned and was one of the longer-running middle school SOPs in San Diego County.

4.1 Event Assignment Methods Comparison

The CVMS SOP has been running for at least 10 years[70], with the SOP being run by parent volunteers for the entire duration. Over that time, there was no consistent organization, rules, or procedures for running the SOP. The volunteer parent leaders (Coordinators) from one year would pick their successor(s) through unknown and varying means[71]. The departing Coordinator(s) typically dropped and ran from the SOP, thus effectively no information regarding the SOP was passed on from year to year[72]. This required the new Coordinator(s)

[68] If the student doesn't think there is a problem, then there really isn't a problem if the problem only exists in the eyes of the parents.

[69] All Events are given the same default rubric weighting. If a coach does not set their Event's rubric by the deadline, then the default rubric is used. The Event rubric lock down was implemented after a small number of Event coaches were found to be modifying their Event rubrics a few days before Regional Team selection was started.

[70] First official recorded appearance in the State competition was in 2006, although CVMS may have participated in SO earlier than this. No records are available to confirm earlier participation.

[71] The SOP Coordinator position was never advertised nor was a general solicitation amongst all CVMS parents ever attempted prior to 2015.

[72] No information or data has been retained by the SOP prior to the 2013-14 school year, and data from the 2012-13 school year appears to have been deliberately deleted apparently to hide evidence of Coordinator bias and favoritism.

every year to create new procedures and practices, resulting in very little consistency from year to year, and not developing any institutional memory about what worked and what did not work.

For the 5-6 years leading up to the 2014-15 school year, the CVMS SOP was limited to ~90 students (6 teams @ 15 students per team plus a handful of alternates), with upwards of 150 students applying to participate in the SOP. Typically, the Coordinators would go through the student applications, accepting students with prior experience in SO and other STEM competitions over students with no competitive STEM experience[73].

At the start of the school year, the Coordinators would assign students to teams and Events, with an emphasis on creating one "strong" team amongst the 6 teams, with the other 5 teams being randomly composed from the remaining students. The students on the strong team were hand-selected by the coordinators based on their limited acquaintance with a small subset of the students in the SOP and the students' applications indicating their prior performance in SO.

In the first years of the CVMS SOP, teams were selected and Events assigned entirely by hand either using paper or a spreadsheet. This manual process was very time-consuming, and the quality of the solution was a function of how much time the volunteers wanted to put into the effort and how well they understood the concepts of problem-solving as it applied to these tasks. The members of Team 1 usually had the best results[74], and the remaining 5 teams often had little correlation between student preferences and which Events they were assigned.

In the last few years leading up to the 2015 season, one parent wrote a Visual Basic (VB) script running in Excel that implemented some sort of a Monte Carlo algorithm to fill out the 5 remaining teams[75]. While this resulted in no change to the way that Team 1 was created, it did result in an improvement for the students on the remaining 5 teams. Unfortunately, the source for the VB script was also lost in the transition from the 2014 to the 2015 season, so it was not possible to use the VB script on any new data to do more comparisons of the old and new Event Assignment algorithms other than on the 2014 season.

4.1.1 Event Assignment: Strategy Proof?

"In game theory, an asymmetric game[76] where players have private information is said to be strategy-proof (SP) if it is a weakly-dominant strategy[77] for every player to reveal his/her private information, i.e. you fare best or at least not worse by being truthful, regardless of what the others do."[78]

Is Event Assignment SP?

---

[73] This had the effect of inhibiting students newly developing an interest in STEM from being able to easily pursue that interest outside of the classroom.

[74] As one would expect, Team 1 was generally the strongest team and it was also composed of the Coordinators' kids.

[75] The VB script assigned students to Events and teams. The VB scripts were not maintained, they did not produce repeatable results and they had ceased to function by the 2014-15 school year.

[76] An asymmetric game is where all players have different strategy sets. In SO, students have different schedule availability and different preferences, so their strategies for achieving their preferred goals are different.

[77] A weakly-dominant strategy is where a strategy B gives at least as good an outcome as choosing strategy A no matter what other players do, and there is at least one set of opponents' actions for which B gives a better outcome than A. https://en.wikipedia.org/wiki/Strategic_dominance

[78] This quote is from https://en.wikipedia.org/wiki/Strategyproofness.

There are three data inputs over which students have control. Therefore there are three possible ways to employ a non-truthful strategy in an attempt to favorably manipulate the system. These are:

1. non-truthful revelation of schedule availability
2. non-truthful revelation of Event preferences
3. non-truthful revelation of the desired number of Events

In the first case, using a non-truthful strategy will yield no better of an outcome than being truthful and often will yield a worse outcome. In the second case, a non-truthful strategy can yield a better outcome than being truthful, but a significantly worse outcome is also very likely. In the third case, a non-truthful strategy can yield a better outcome and may also cause significant detrimental effects on Event Assignment and Team Selection. While Event Assignment is not perfectly SP, it is possible to minimize the incentive and opportunity for participants to benefit from being non-truthful.

These three cases are examined in more detail herein.

4.1.1.1 Gaming via Availability

The following discussion is based on actual results from all 6 years of performing Event Assignment, where some years displayed the Event schedule, but in most years, most schools hid the schedule.

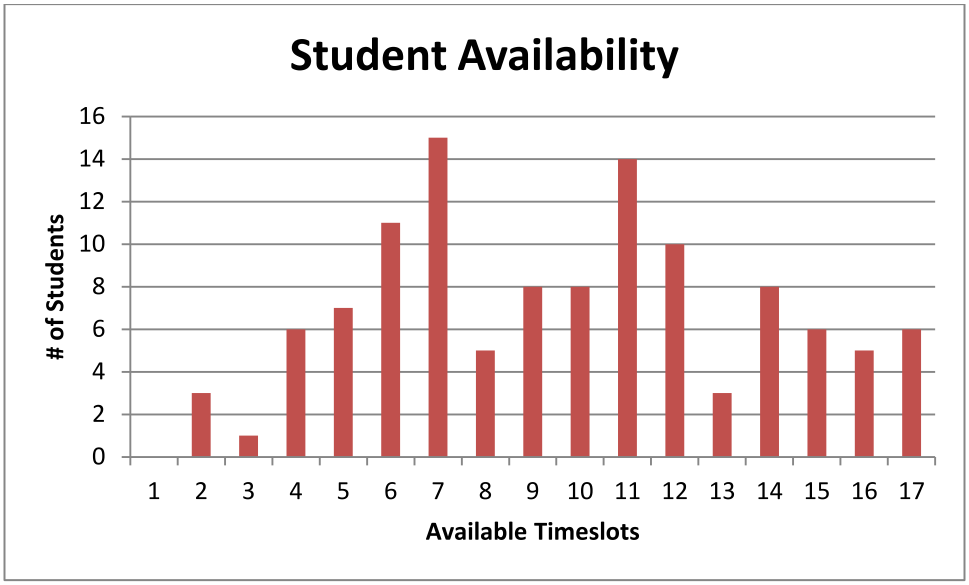
Despite the advice provided to families regarding truthful revelation of Availability, every year 5 to 10%[79] of the families decide that they can game the Event Assignment software. These families do this no matter in which mode the registration software is run—although the Hidden mode requires much more groundwork on the parents' part to investigate which Events are likely to be offered in which timeslots. Anecdotally, there appear to be fewer parents who try to game the system when Events are hidden, probably in part due to the extra effort required to investigate the schedule. Primarily, the parents believe that if they only list those timeslots that match their student's top *n* Event choices as being available, then their strategy will force their kids to get only their top Events no matter what. This may seem intuitively true—except it is false when:

- an Event fills before that student comes up for assignment, or
- there is a test conflict with two or more of those preferred Events, or
- there is more than one Event scheduled to be taught in one or more of the student's available timeslots.

If one or more of these conditions occur, the student will end up with fewer Events than desired or perhaps be assigned an Event which was much lower on their preference list than they would have gotten if they had been truthful in listing their timeslot availability. Note that if none of these conditions occurs and the student is assigned their top 3 Events, then the algorithm would have assigned the student those 3 Events even if had they truthfully revealed their preferences and all of their availability. Therefore, a student will *never* be worse off by truthfully revealing their availability.

[79] Table 12: shows there are 10 students with 4 or fewer available timeslots and 7 more students with 5 available timeslots. These 17 students (out of 116) have indicated that they have less than 1/3 of the possible timeslots available. It is likely that many (or most) of these 17 students are attempting to game the Event Assignment by limiting their actual availability.

Table 12: Student Availability



Invariably, many of those parents, whose attempts to game the system backfire, subsequently want to "correct" the outcome. After Event Assignment is done and the results are known, those parents demand that their students be placed in other Events (now full) in timeslots they hadn't originally marked as being available. While there are usually a handful of Events that still have seats still available after Event Assignment, those Events are often lower on their kid's preference list than the full Events and thus 'unacceptable' to those parents.

In trying to game the system by 'forcing' the RSD to work with a limited number of timeslots for a student, at best, the student does as well as if they were truthful in their input, but it is not unusual that they are worse off. Unfortunately, there is no way to look at the data to determine what percentage of participants are untruthful and how many of them ended up worse off than if they had been truthful. While some parents self-identify when they demand re-assignment, other parents accept the result as-is. And very few parents are likely to answer a survey question on whether or not they tried to game the system—let alone answer the survey truthfully!

4.1.1.2 Gaming via Event Swap

Another possible gaming strategy is to swap the order of a student's preferences if the family believes that an Event at the top of their preference list is not as popular as an Event a little lower down on their list. For example, if a student lists Events A, B, and C in order as their top 3 choices, but believes that C will be popular and subject to filling to capacity before the 3rd round of Event assignment, and also believes that A is not very popular and unlikely to fill before the 3rd round. Then intuition might suggest that swapping the preference order to C, B, and A will improve the chances of being successfully assigned to all three Events rather than getting A and B and missing C.

No information is provided to any stakeholder about the demand for Events during the registration period, and the popularity of Events can vary greatly from one year to the next based on which Events are offered, who is coaching an Event, the student pool's interests, etc. Therefore, any beliefs about which Events are in greater demand are almost entirely speculative.

The risk is that if the family's speculation is incorrect, then the student may miss out on getting their most preferred Event in order to get a lower-ranked Event. In addition, if the family is not cognizant of test conflicts and teaching conflicts, they risk not getting their most preferred Event because it has a conflict with their lower-ranked Event.

Thus, when trying to game the system by swapping Event preferences based on speculative assumptions about demand for certain Events, the outcome is highly dependent on how other students rank their Event preferences. If the speculation is correct, the swap may yield a better result. If the speculation is incorrect, the swap is likely to yield an inferior result. And if the Event conflicts are a factor, then the swap will *always* yield an inferior result.

Though Event Assignment may not be perfectly SP with respect to preference swapping, to the best of my knowledge during the previous 6 years, no one has ever successfully executed a preference swap. Typically, these successful strategies quickly spread by word of mouth in the parental community and eventually get reported to me: no such report has occurred. If an attempt was made and failed, then it is possible that the family requested modifications after Event Assignment was completed without mentioning that their reason for needing a modification was due to a failed gaming attempt.

Also note that during the previous 6 years that no Event filled to capacity in the first round of Event Assignment. Thus, all students were assigned to their first *available* Event choice. Some Events *did* fill (sometimes early) in the second round, so some students who swapped preference order may have missed out on their higher preferred Event in the second (or subsequent) round of assignment due to their swapping the order of their Event preferences. Due to the fact that Events are unlikely to fill in the first round, the risk/reward for trying to game the system by swapping is much *less* favorable than it is at HBS. [6]

Table 13: Student Availability by Time Slot

Event	Time Slot	Students Available	% Students
MICR	0	72	62%
ECO	1	61	53%
SOL	2	72	62%
ANAT	4	58	50%
DP	4	58	50%
CB	5	68	59%
DD	5	68	59%
HOVE	5	68	59%
FF	6	73	63%
HERP	6	73	63%
WIDI	7	64	55%
ROLL	8	75	65%
EXPD	9	73	63%
ROCK	10	56	48%
RS	11	61	53%
TOW	12	68	59%
BUGG	13	57	49%
MET	13	57	49%
POTP	14	59	51%
WS	14	59	51%
MARC	20	55	47%
THER	21	70	60%
OPTIC	23	77	66%

56.3%	Average
58.6%	Median
66.4%	Max
47.4%	Min
0.0597	Std Dev

Table 13: shows a typical distribution of student availability by timeslot. This example from CVMS in 2018 covers 116 students requesting 380 total Event seats, which equates (per equation (3)) to a capacity of 16.2 students per Event (24.3 students for a 3-student Event).

The table tallies the number of students marked as available in each timeslot for which an Event is taught. This represents the theoretical maximum number of students that could enroll in each Event. These tallies exceed the maximum capacity allowed for each Event. Thus, the tallies alone cannot prove that an Event will not fill in the first round of Event Assignment.

The colored rows indicate that there are two or more Events taught in a timeslot. For these Events, the students are distributed amongst the Events in the same timeslot. This further reduces the theoretical maximum number of students that could enroll in those Events. However, even with this dilution, the number of students available for each Event still exceeds the calculated Event capacity.

Table 14: Student Event Preferences by Rank

Event	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
ANAT	15	1	3	6	6	10	6	6	8	2	7	4	5	3	5	4	4	4	4	4	3	2	4
BUGG	4	7	8	6	5	5	4	3	4	3	5	6	3	2	3	7	5	4	6	9	6	3	8
CB	10	11	2	14	9	10	4	3	5	3	3	3	6	6	3	1	3	2	5	4	4	5	0
DD	5	7	9	6	5	5	8	2	8	3	4	3	4	6	7	9	4	3	2	5	2	5	4
DP	4	1	2	1	2	5	3	8	7	3	9	2	7	5	6	7	6	6	9	4	6	6	7
ECO	0	0	2	6	5	1	10	3	4	3	9	4	5	4	3	6	11	12	4	8	5	9	2
EXPD	8	5	3	6	3	1	3	6	3	7	4	7	6	10	6	5	4	5	6	1	6	4	7
FF	7	3	4	3	7	5	4	7	6	4	5	6	5	5	3	4	6	2	3	6	5	7	9
HERP	1	5	3	3	1	3	3	2	3	2	3	9	5	10	7	8	4	7	8	7	4	7	11
HOVE	5	8	4	6	6	2	7	7	4	3	3	4	3	3	3	1	7	6	5	5	11	9	4
MET	0	4	1	0	0	8	4	7	7	11	6	5	9	7	10	5	8	6	4	6	1	3	4
MICR	2	6	2	5	8	5	6	5	7	10	5	4	6	2	9	5	3	1	7	7	6	4	1
MARC	2	6	5	5	6	7	8	8	3	4	3	9	2	5	3	6	5	5	9	4	4	5	2
OPTI	7	10	7	4	3	6	2	3	6	8	7	5	5	4	6	2	7	5	5	1	3	1	9
POTP	6	6	18	4	7	8	5	7	7	5	4	4	5	5	6	3	2	3	2	2	6	1	0
RS	1	2	4	1	2	3	6	6	4	4	4	3	8	2	6	7	4	5	4	7	13	7	13
ROCK	3	6	4	4	4	5	2	1	4	8	8	6	8	10	8	7	5	2	8	4	6	1	2
ROLL	6	6	2	5	8	3	4	8	7	4	5	3	7	6	4	6	2	6	5	4	1	9	5
SOL	11	4	10	6	8	5	8	9	4	9	4	4	5	3	2	2	5	3	2	7	3	1	1
THER	2	3	4	2	4	3	4	3	3	3	6	9	5	6	4	5	12	11	2	6	7	7	5
TOW	7	8	9	11	6	5	6	2	8	8	2	4	3	4	3	7	2	3	5	4	4	2	3
WS	7	5	5	7	6	4	6	5	2	4	5	8	3	4	3	4	4	4	8	7	5	7	3
WIDI	3	2	5	5	5	7	3	5	2	5	5	4	1	4	6	5	3	11	3	4	5	11	12

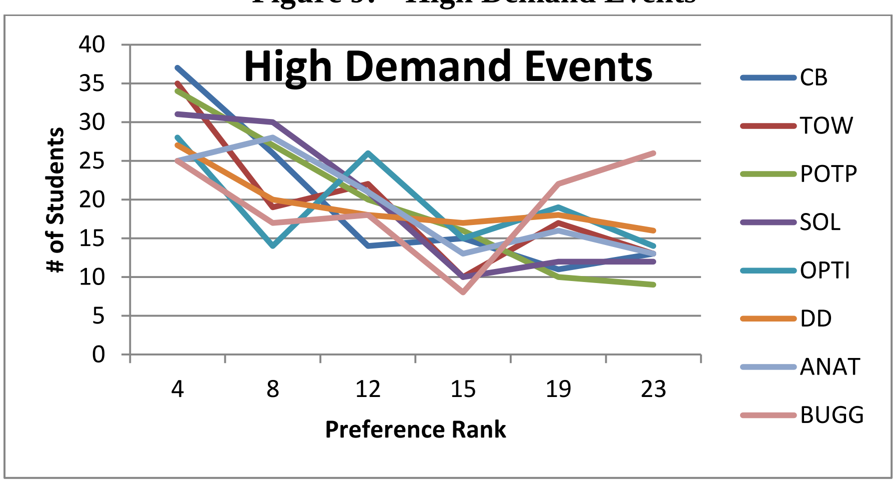
Table 14: shows the number of students who ranked each Event at each of the 23 possible levels of preference. In this table, there are very few cells[80] where the tally of students exceeds the

[80] In fact, there is exactly one cell that exceeds the maximum capacity: POTP rank 3 @ 18 students. This is a typical result every year.

capacity of an Event. Based on Table 13:, at least 1/3 of the students[81] won't be available in the POTP timeslot, which statistically reduces the number of students that can legally be assigned to POTP at each preference rank to a level below the maximum capacity of POTP. Specifically, looking at the row for POTP in Table 14:, the maximum demand for POTP is 18 students in preference rank 3—which is also the greatest demand for any Event in any preference column in this table. Decreasing that demand by 1/3 leaves a demand of 12 students in that cell who are likely to be available to be assigned to POTP, where 12 students is less than the calculated 16.2 student capacity for the Event. This example is simply to reinforce the idea that in this domain of SO, it is highly unlikely that an Event would fill to capacity in the first round of Event Assignment.

Further, Table 14: shows the widely varied interests of students across Events. As shown in Figure 9:, Figure 10:, and Figure 11:, there are certainly Events which are more popular than others, but there are no Events which are overwhelmingly popular or underwhelmingly unpopular. But the popularity of Events changes from year to year due to the changing student pool and due to the different mix of Events. Just because an Event was popular last year does not mean it will be popular this year. Also, certain Events are mutually exclusive: if a student is assigned an Event in one round of Event Assignment, then in subsequent rounds that student can no longer be assigned to any Event which has a conflict with Events the student already has. The changing popularity of Events and Event conflicts are both elements that are different from HBS and work against someone trying to game the system in the same manner that students do at HBS. Specifically, students in SO do not know the popularity of Events: popularity of Events can change significantly from one year to the next and there is no way to extrapolate popularity of Events based on prior year demand since that information is not released. Also, if students are unaware of the Event conflicts, they can put themselves into a position where they *exclude* themselves from a top choice Event through swapping preference positions with one of their lower preferred Events which has a conflict with their top choice Event (which cannot happen at HBS).

Figure 9: High Demand Events



[81] From the statistics on Table 13:, the maximum students available in any timeslot is 66.4%, which means that no fewer than 33% of the students are unavailable in any timeslot. In the specific timeslot for POTP, 49% of the students are unavailable in that timeslot—which is greater than 1/3 of the student pool.

Figure 9:, Figure 10:, and Figure 11: condense the student preference demand in Table 14: by summing 4 preference rank columns at a time and plotting the results. The condensed tallies are sorted to divide the 23 Events into 3 groups: High Demand Events (8 Events), Middle Demand Events (8 Events), and Low Demand Events (7 Events).

In Figure 9:, the more popular Events trend from more demand on the highly ranked preferences at the left side of the plot to low demand on the low ranked preferences on the right side of the plot. Since there are 116 total students who rank each Event once, if a large number of students rank the Event highly, there are not many students out of the 116 who can rank the Event low. Thus, a popular Event can be defined as an Event that is ranked highly by more students than students who rank it low.

Figure 10: Middle Demand Events

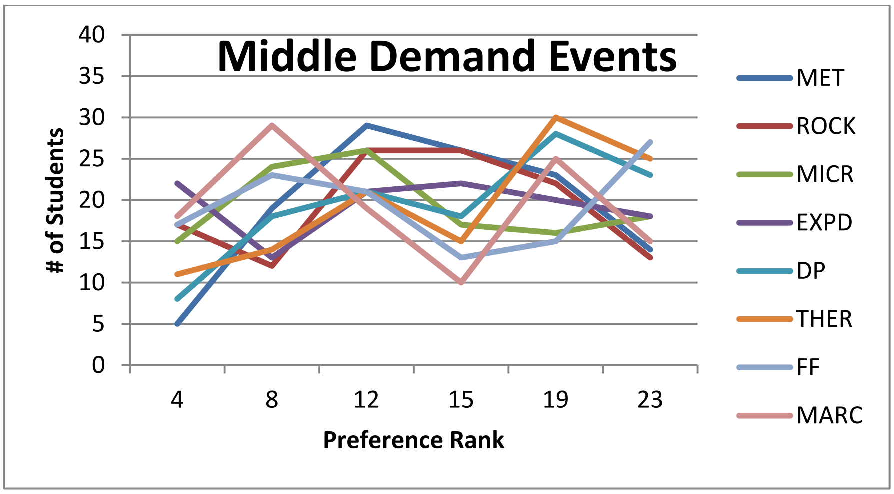


Figure 10: shows the middle demand Events. These Events tend to have relatively uniform demand across all preference ranks. These Events may also have relatively low demand at the high preference ranks and relatively low demand at the low preference ranks, yielding an upside-down U-shaped curve. These Events may also have somewhat higher demand at the higher preference ranks and at the lower preference ranks, with a higher demand in the middle ranks, which makes a U-shaped curve.

Figure 11: Low Demand Events

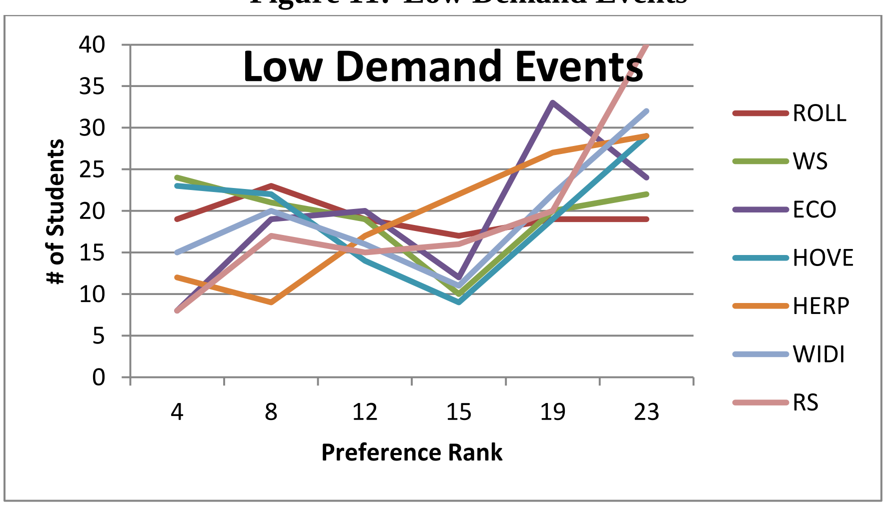


Figure 11: shows the low demand Events. These Events have a low demand in the higher preference ranks and high 'demand' in the low preference ranks. This yields a generally upward sloping demand curve from the left side of the plot to the right side of the plot. Note that even these 'unpopular' Events have their student fans, as all of these Events easily have at least 16 students who ranked them within their top 8 Event preferences. That is to say, it is not a difficult task to fill these Events with a number of students who are highly enthusiastic about the Event.

Obviously, if this demand information was provided to the students, they would be much more inclined to attempt to game the Event Assignment by swapping Event preferences based on the popularity of Events. As with HBS, there would be artificially higher demand for Events perceived to be popular. However, this would be a very dynamic situation as students would react to changing popularity[82] by changing their preference rankings, which would change this data, which would cause the students to adjust again. The result from making this demand data available would not be a positive improvement over the current system.

4.1.1.3 Gaming via Event Count

Usually it is announced in advance whether or not there is an incentive preference for students who sign up for more than 3 Events. There are two possible gambits here:

1. A student who wants 3 (or fewer) Events would sign up for 4 Events but only mark three (or fewer) timeslots as being available. Since a student cannot be assigned more Events than they have timeslots available, the student gets the Event Count benefit without having to do 4 Events.
2. A student would sign up for 4 Events with the intention of immediately dropping their 4th Event after Event assignments are announced.

Both of these gambits have been previously exploited during the registration process. This is known to be true since boasting of these successful gambits circulated amongst the students and parents and these exploits were subsequently reported to me.

The first gambit was easily solved with a simple software fix. When the Event Assignment software is kicked off, the student's reported number of available timeslots is compared to the number of Events they have requested, and the lower of those two numbers are used during the Event Assignment for that student. This prevents a student from benefiting from the 4-Event incentive preference if they, in fact, cannot be assigned 4 Events.

The second gambit cannot be prevented with software, nor can it be prevented by instituting a rule that says "A student may not drop their 4th Event" or a more general rule of "A student may not drop any Event". Because the SOP is in a public school system, the State laws are such that students cannot be 'forced' to keep all their extracurricular activities. Even if the SOP were a private organization, it is impractical to try to enforce a "no-drop" restriction as there are too many external circumstances that can change for a family (e.g., a family moves out of the area and is no longer able to attend their Event classes).

---

[82] Since students register on a random arrival basis over a one week period, the demand for Events changes for two reasons: as more students register overall demand goes up, and as students who are already registered modify their preferences based on the evolving demand pattern. While it is possible this might reach equilibrium within a 7 day registration period, there is no guarantee it would reach equilibrium every year. And, based on the HBS experience, the artificially higher demand for some Events would exceed the capacity of those Events, resulting in a random assignment of seats amongst those students rather than assignment based on actual student preferences.

There can be significant negative effects that result from students purposefully dropping their 4th Event. One is that the number of students in the Events becomes unbalanced, which will later constrain Team Selection. A second is that those additional seats being temporarily 'occupied' may have prevented other students who wanted those seats from being assigned those seats and instead being assigned to a different Event lower on their preference list. Thus, the MI of some individual students is made worse, as well as worsening the overall student pool's MI.

The only practical solution to these gambits is *not to use* the 4-Event incentive option. While it is necessary to have some 4-Event students in order to fully staff a team, there appears to be sufficient incentive for students to sign up for 4 Events. There is a mathematical advantage[83] that a 4-Event student has over an otherwise equivalent 3-Event student for being assigned to a higher scoring team (and likewise the State team). Also, a student's enthusiasm for SO, STEM, learning, and taking on challenges provides incentive for students to take on 4 Events.

Similarly, the 2-Event penalty incentive may cause students who are only interested in participating in 1 or 2 Events to register for 3 Events with the intent to drop Events after the Event Assignment is complete. This behavior has an even larger potential impact on both team selection and the MI than does the 4-Event incentive option. This is because the behavior affects Event Assignment in the 2nd and 3rd rounds of assignment (with ripples into the 4th round) rather than only having an impact in the 4th round. In addition, in the last 2 years of the SOP, there are more students requesting 1 or 2 Events than those requesting 4 Events, so there are likely more 1- and 2-Event students using this gambit than 3-Event students using the 4-Event gambit.

This results in the recommendation to head coaches that they do not use either the 4-Event incentive preference option or the 2-Event penalty option for Event Assignment.

4.1.2 Event Assignment: Envy Free

"Envy-freeness (EF) is a criterion of fair division. In an envy-free division, every agent feels that their share is at least as good as the share of any other agent, and thus no agent feels envy."[84]

Is Event Assignment using the RSD Envy Free?

The Event Assignment is *not* Envy Free, as people do not see/understand that schedule and conflict constraints should have a bearing on their outcomes vs. the perceived outcome of others. Every year, 5 to 10% of the families ask for details about why their student did not get their top 3 Events. They ask because they know there are other students who did get their top 3 Events, and want to know why they didn't get a similar result. The reasons, in order of likelihood, come down to: first, the timeslot availability of the student; second, Event testing and teaching conflicts; and third, the Event was filled.

---

[83] 4-Event students are required on every fully-staffed team. 4-Event students can also fulfil the role of a 3-Event student by dropping an Event when assigned to a team. A 4-Event student can be slotted into a team as a 4-Event student or a 3-Event student using any 3 of their 4 Events. A 3-Event student can only be slotted into a team with their 3-Events. Given two equivalent students except that one student has 4-Events and the other 3-Events, the 4-Event student has a greater probability of being assigned to a team before the 3-Event student because the 4-Event student has more potential configurations that can be fit in a team than does the 3-Event student.
Note also, both the 3- and 4-Event student could also be slotted into the team using only 1 or 2 of their Events, but even in that case, the 4-Event student has more possible configurations for slotting than does the 3-Event student.

[84] https://en.wikipedia.org/wiki/Envy-freeness

Most families involved in SO do not pay attention to test conflicts and are surprised to learn that that was why they did not get a particular Event. Likewise, many families appear to be surprised to hear that they did not get an Event due to their limited schedule availability, despite having been informed of these factors prior to registering for SO, and during the on-line registration process.

Thus, the fairness of the Event Assignment is judged by a portion of the stakeholders based on the ordinal ranks of the Events that were assigned to their student vs. the ordinal ranks of the Events assigned to the students who received their top choice Events. They give no consideration to the effects of availability or conflicts on the Event Assignment outcome.

4.1.3 RSD vs. VB and FCFS Algorithms

In the comparison of the new RSD Event assignment algorithm versus the old VB Monte Carlo algorithm, there were 104 students in the RSD pool vs. 89 students in the VB pool. For the Event assignment results comparison between VB and RSD, it was only possible to run a direct comparison using data from the 2013-14 school year[85].

For the 2014-15 school year, there were 50% more students[86] participating, with 1 additional Event requiring 3 students, and 23 students requesting four Events[87]. Additionally, students marked an average of 11.95/18 (66%) time slots available in 2013-14 vs. 11.4/23 (50%) time slots available in 2014-15[88]. These differences in the 2014-15 data produced a more constrained and difficult optimization problem than the prior year's data. The 2014-15 data is compared using the RSD against a First-Come, First-Served (FCFS) assignment methodology used by some local area schools.

The 2014-15 data was selected for comparison with the 2013-14 data because it is most similar in circumstances with the 2013-14 data. Approximately half of the students in the pool were involved in both years (the other half of students moved on to high school and were replaced by students moving up from elementary school), and the list of Events between the two years were substantially similar[89]. As described in the prior paragraph, the primary differences were the student pool being 50% larger and the pool had less time slot availability.

With FCFS, it is very similar to a single round RSD, where the student being processed selects *all* of their Events before the next student is processed. As described by [17]: "FCFS is simple and the most widely used assignment rule in practice, but it leads to inefficient outcomes and envy in the allocation." The inefficiency comes about because who gets what is based entirely on order of arrival and not preference or intensity of preference. The envy in the allocation comes from the fact that the assignment is nowhere near equal or equitable: FCFS provides very

---

[85] There is no data from years prior to 2013-14. The student preference data from 2014-15 could not be run on the VB program as it stopped functioning—the VB program would no longer run even on the same computer with the same data as it was used in 2013-14. The VB program was over 6 years old, and the original programmer(s) are unknown (and likely wouldn't be interested in troubleshooting the program even if they could be found).

[86] There were 135 students in 2014-15 versus 89 in the VB data from 2013-14.

[87] 11 of 90 (12%) students in 2013-14 were assigned 4 Events versus 23 of 135 (17%) students being assigned 4 Events in 2014-15.

[88] 2014 students marked an average of 11.95 of 18 timeslots available versus 2015 students marking an average of 11.4 of 23 timeslots.

[89] Recall that approximately 6 of the 23 Events are replaced each competition year.

good results for the first students to arrive at the queue, and generally much poorer results for the students who arrive last at the queue (since they must pick only from whatever is leftover). The students in the middle of the queue get results that are then at various levels between good and poor. This results in a much wider range of MI amongst the students. These effects are seen in the figures and tables in the following subsections.

In the actual real-world implementation of FCFS for Event Assignment, there is random arrival of students at a web page, where they hurriedly claim their Events. The students are in a hurry since there is no gating to allow one student to complete their selections before the next student is allowed to make their selections. Thus, multiple students can be overlapping and vying for Events simultaneously. It is not unusual in these situations that the students with faster Internet connections and better 'mousing' skills also grab better assignments simply because their browser page refreshes faster than other students' computers can.

It is important to note that to arrive at the most favorable results with the FCFS methodology, 10,000 iterations were run and the results with the best MI and lowest total Event shortage is used. Unlike the RSD where it is practical and makes sense to run multiple iterations to arrive at a more optimal solution, in the real world with a true FCFS scenario, it is impractical to run multiple iterations as that would require the participants to register and select Events multiple times over multiple days.[90] However, it is possible to use a randomized FCFS algorithm applied to the data collected during student registration to compare the best possible results from FCFS versus the best possible results from RSD[91].

It is also important to note that the FCFS is the *only* algorithm of the three algorithms that left 10 students short of their desired number of Events *and* 4 additional students with *no* Events at all. This represents 10% of the student pool (14 out of 135 students). And this occurred even *after* running 10,000 iterations and taking the *best* solution out of all the solutions. This fact alone is sufficient to declare the FCFS method unacceptable, especially in the public domain of the California educational system.

4.1.3.1 Highest Rank Event Assigned

Figure 12: shows the number of students who were assigned their *n*th ranked Event as their highest ranked Event. Adding the first three bar columns together, this shows the number of students who were assigned *at least* one of their top 3 Event choices. From Figure 14: the first bar column shows the number of students who were assigned all three of their top Events. Table 15: combines these statistics, and also adds the percentage of students who were short 1 or more Events and also the number of students who were not assigned any Events.

In Table 15:, it can be seen that the RSD consistently outperforms both the VB and FCFS algorithms. The percentage of students being assigned at least one of their top 3 choices and the percentage of students being assigned their top choice Event is both significantly and consistently better with the RSD. While FCFS does better with the percentage of students getting all 3 of their top 3 choices in the 2015 data, the trade-off is a nearly equal percentage of students who are short Events or get no Events at all. This effectively nullifies the "All Top 3"

[90] Not to mention that the participants would be upset if the final allocation was one in which they did not receive their personal best assignment out of the multiple iterations that were conducted.
[91] A real-time FCFS result would likely fair much worse in comparison with the RSD as there would only be one run and it is highly unlikely that that one run would be an optimal solution.

advantage enjoyed by FCFS. Neither of the VB or RSD algorithms left any students short Events, let alone without any Events at all.

Table 15: Percentage Students Assigned Top Choices

Method	At Least 1 of Top 3	Top Choice	All Top 3	Short 1 or 2	No Events
RSD	94%	70%	13%	0	0
VB	90%	50%	3%	0	0
2015 RSD	93%	65%	7%	0	0
2015 FCFS	81%	56%	12%	7%	3%

The distribution of highest rank Events for the three algorithms is also telling. In Figure 12:, the tail on the RSD assignment had a steeper decline than the VB assignment's long tail. In the case of RSD, the student receiving their 9th highest preference Event was more due to limited schedule availability of the student[92] and not the algorithm. With the VB algorithm, the student with their 9th highest preference was more an artifact of the algorithm, and not because of the student's limited availability[93]. With the 2015 data for RSD, the student receiving their 11th highest Event was due to limited schedule availability of the student[94]. For the FCFS, the two students receiving their 11th highest Event occurred due to limited schedule availability and the order in which they were processed, and the student receiving their 10th highest choice was due entirely to their order in which they were processed[95]. Note also that FCFS has 4 students with no Events, and this fact is not represented in the figure.

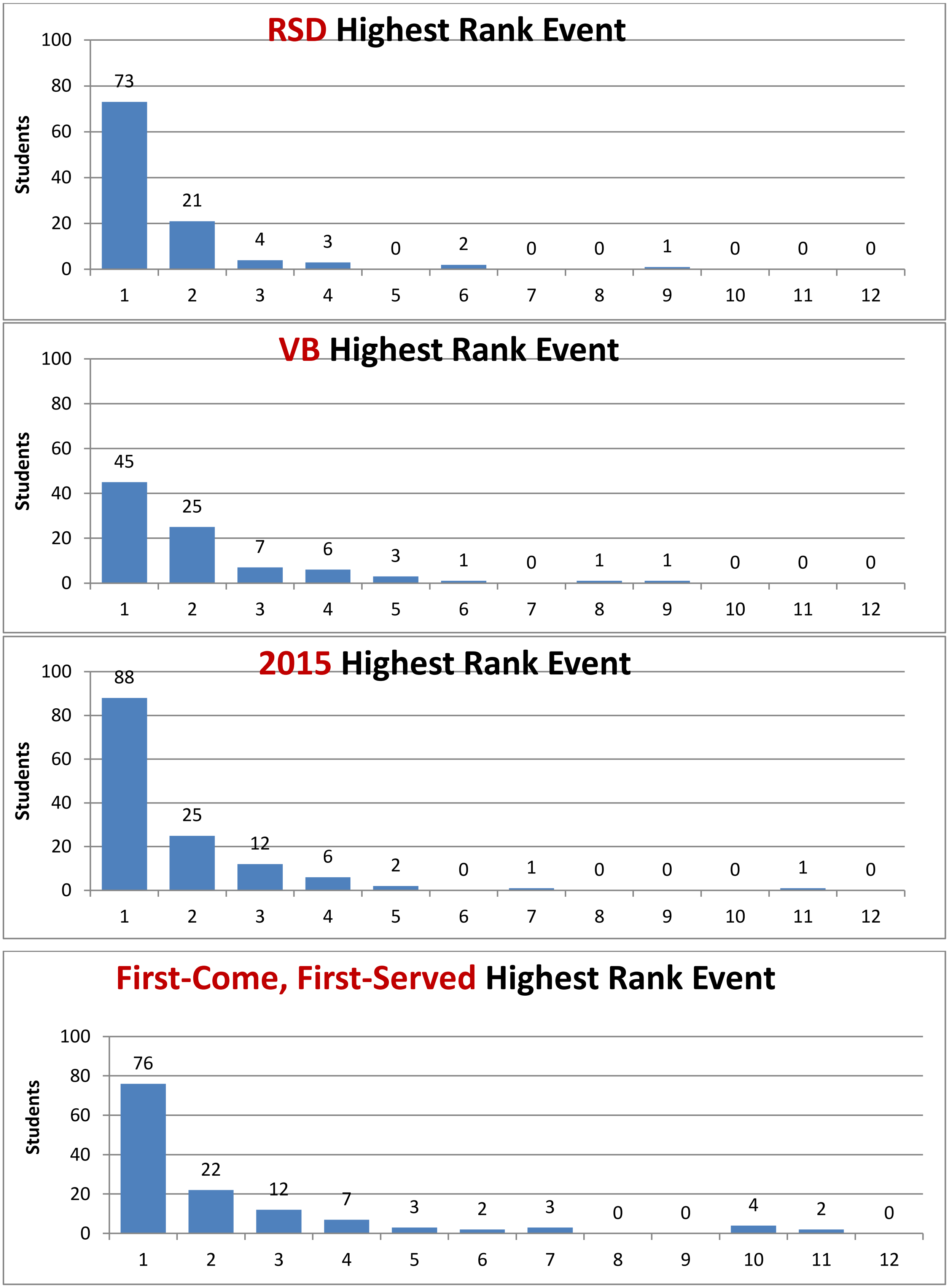
Table 16: also shows that both the RSD and 2015 Event Assignment solutions are superior to the VB and FCFS results. RSD and 2015 both have lower Averages and smaller standard deviations than VB and FCFS. While 2015 RSD and FCFS have rather similar profiles on the left side of the plot, FCFS has a grouping on the right side of the plot which is much larger than the RSD's outlier. This confirms what the bar graphs show visually: RSD and 2015 results are clustered closer together and skewed closer to the origin of the plot than VB and FCFS.

Table 16: Highest Rank Event Statistics

	RSD	VB	2015	FCFS
Maximum	9	9	11	11
Minimum	1	1	1	1
Average	1.54	2.00	1.67	2.24
Median	1	1	1	1
Std Dev	1.19	1.51	1.32	2.24

[92] The student had marked 4 of the 19 timeslots as available.
[93] The student had marked 12 of the 18 timeslots as available. Note also that these were two different students.
[94] The student had marked 8 of the 23 timeslots as available.
[95] These 11th choice students had marked 4 or 6 of the 23 timeslots available. The 10th choice student had 12 timeslots marked as available, which is above average out of the student pool for availability.

Figure 12: Highest Rank Assigned Event



4.1.3.2 Lowest Rank Event Assigned

Figure 13: is a bar chart showing the lowest ranked Event assigned to each student. The chart shows the RSD algorithm grouping the vast majority of students to the left side of the graph, and the VB algorithm being a fairly uniform distribution across the entire width of the graph. The average Low Event score for each of the two assignment algorithms shows the VB algorithm to be nearly twice as bad off as the RSD algorithm. Additionally, the absolute number of students assigned at least one Event lower than their 12th preference choice for the VB algorithm is 37 versus 7 for the RSD algorithm.

The 2015 data shows 32 students with their lowest preference Event lower than 12th choice. The graph shows the bulk of assignments on the left side of the graph, tailing off to the right side of the graph. The higher number of low ranked student Events on the right side is due primarily to limited availability of the students[96] in that region of the graph. FCFS shows 18 students with their lowest preference Event lower than 12th choice, however, there are also 10 students who are short Events and 4 students with no Events. The distributions of 2015 RSD and FCFS are surprisingly similar from 3rd to 12th choice, after which FCFS remains consistently lower (which is primarily due to the 14 students with missing Events).

The tail on the RSD assignment also had a steeper decline than the VB assignment's long tail. In the case of RSD, the two students receiving their 16th or higher preference Event was due to limited schedule availability of those students[97] and not the algorithm. With the VB algorithm, the 28 students with their 16th or lower preference was primarily an artifact of the algorithm, and not because of the student's limited availability, since we know the RSD was able to do significantly better using the same data and constraints.

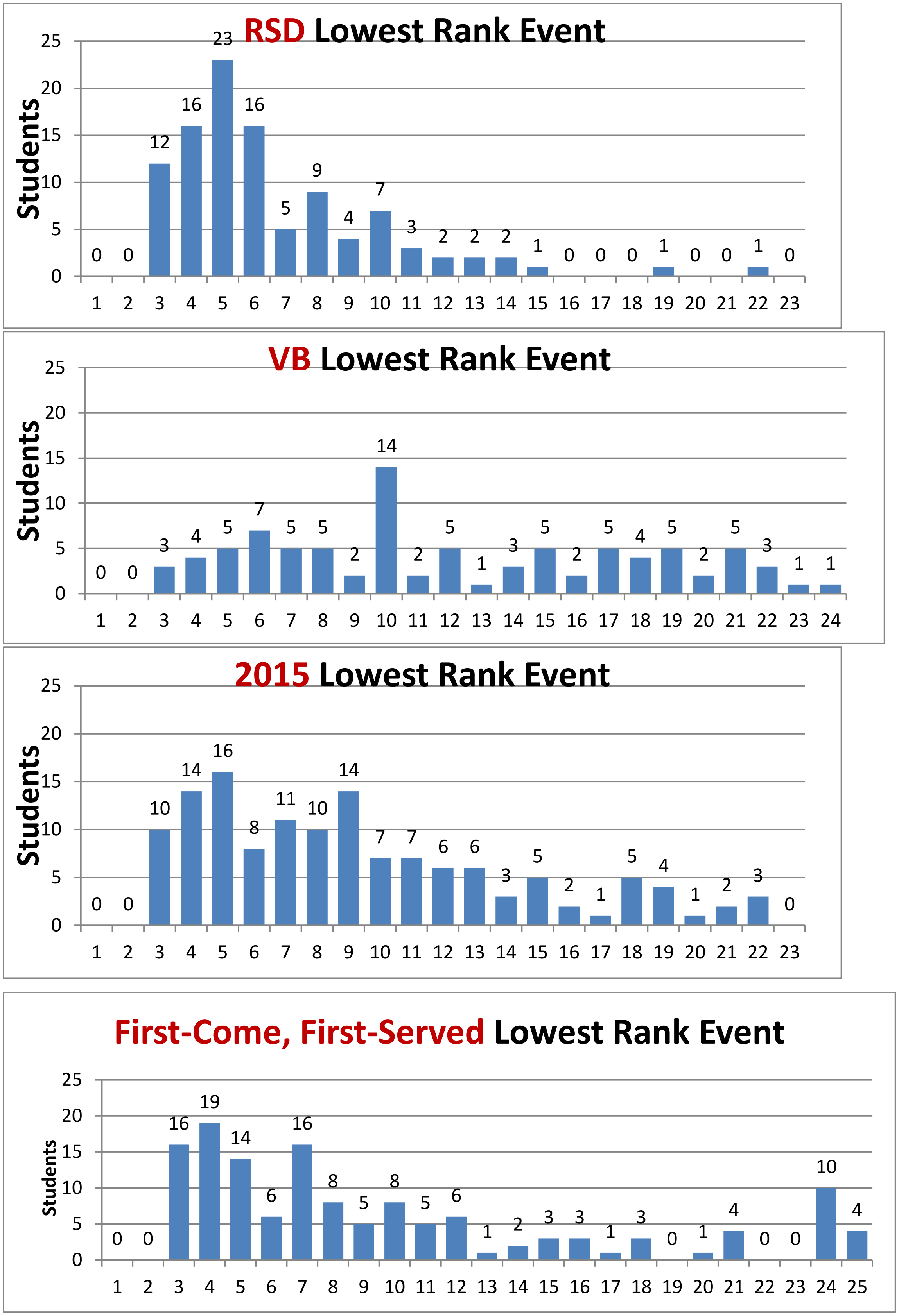
Table 17: Lowest Rank Event Statistics

	RSD	VB	2015	FCFS	FCFS Penalty
Maximum	22	24	22	21	24
Minimum	3	3	3	3	3
Average	6.64	12.12	9.29	7.99	9.65
Median	6	10	8	7	7
Std Dev	3.42	5.78	5.02	4.71	6.61

[96] These are primarily students with 4 to 8 timeslots of availability.

[97] The student had marked 4 of the 19 timeslots as available.

Figure 13: Lowest Rank Assigned Event



Note that students opting for a 4th Event do not have the preference numbers for those 4th Events included in this graph. That is because adding a 4th Event, by definition, will produce a poorer preference value: 4th Events are assigned only *after* all students have been assigned their 3rd Event, thus many Events will already be full; 4th Events are optional, and only 12 students opted for a 4th Event.

Note also that while there are a total of 23 Events, one student in the VB results is shown as having an Event ranking of 24. This is because the student did not rank all 23 Events and was assigned an unranked (slack) Event by the VB algorithm. This also happened with two students who did not rank all 23 Events and were assigned a 4th Event that they did not rank (these two students had requested a 4th Event). The RSD algorithm did not encounter this problem. Those students who did not rank all of the Events were assigned Events without reaching the end of their preference lists. Since the RSD algorithm is more efficient than the VB algorithm, it did not need to go as deep into the students' preference lists to complete the Event Assignment.

Table 17: also shows that both the RSD and 2015 Event Assignment solutions are superior to the VB results when it comes to the last (and lowest preference) Event assigned to students. RSD and 2015 both have significantly lower Averages, Medians, and smaller standard deviations than VB. This confirms what the bar graphs show visually: RSD and 2015 results are again clustered closer together and skewed closer to the origin of the plot.

For RSD 2015 and FCFS, Table 17: shows that FCFS bests RSD in average, median, and standard deviation. However, these numbers (again) do not take into account the 14 students the FCFS left short Events or with no Events. Other than adding an arbitrary penalty to FCFS for those students (such as assigning a 24 for each missing Event), it's difficult to assess a mathematical cost and effect on these statistics. Including the arbitrary penalty of 24 in the data for students missing Events shows FCFS to be much less desirable than RSD.

4.1.3.3 Student Misery Index

Figure 14: illustrates the MI. To make the cleanest comparison between the VB and RSD algorithms and the 2014 and 2015 data, only the first 3 Events of students were included in the data. Students who had 4 Events had their 4th Event excluded from their calculation. The Average MIs for each of the 4 sets of data are:

- RSD: 3.87
- VB: 6.63
- 2014-15: 4.96 (including students' 4th Events: 5.45)
- FCFS: 5.23
- FCFS Penalty: 6.24

The VB algorithm's Average MI is 71% higher than the RSD's MI. Additionally, the RSD algorithm's distribution is concentrated to the left side of the bar chart, while the VB algorithm is much more uniformly distributed across the width of the graph.

Note that with the RSD, the few outliers on the right side of the graph are there due to the limited schedule availability of those students.

Also note that in the VB algorithm, the 3 students who were assigned all three of their top 3 preferred Events were the children of the three SOP Coordinators. It is not possible to determine if the VB algorithm reached this solution naturally[98], or if the solution was influenced by the individual running the VB program. However, the probability of this particular coincidental outcome is extremely small.

The numerically higher Average MI for the 2014-15 school year is explained in part by the lower overall indicated availability of students. The RSD/VB data set from the 2013-14 school year indicated the average student availability to be 65% of the available time slots, whereas the 2014-15 average student availability was 50% of the available time slots. This may in part be an artifact of students (parents) trying to 'game the system' by marking their availability based on when they believed certain Events were scheduled[99]. This is also in part due to a slightly higher degree of *Event conflict* in the 2014-15 competition schedules versus the 2013-14 competition schedules because the San Diego Regional test schedule differed from the National test schedule creating more test conflicts between Events and imposing more constraints on the system.

Even with a higher overall MI for 2014-15, the MI value still beat the VB algorithm by 31%.

Comparing the 2015 RSD and FCFS, RSD bested the (un-penalized) FCFS in MI by 5% and bests the penalized FCFS by 26%. In the plots, RSD and FCFS appear fairly close in distribution to a total MI of 14, after which the RSD results are more concentrated toward the left side of the plot and the FCFS looks more like a uniform distribution. Throwing in the short and no-Event students using the penalty values, then FCFS looks far worse with 14 students spread out along the right side of the plot stretching out to a total MI of 72.

Table 18: also shows that both the RSD and 2015 Event Assignment solutions are superior to the VB and FCFS results when it comes to the complete solution set. RSD and 2015 both have significantly lower Averages and Medians, as well as significantly smaller standard deviations than VB. This confirms what the bar graphs show visually: RSD and 2015 results are again clustered closer together and skewed closer to the origin of the plot resulting in superior solutions.

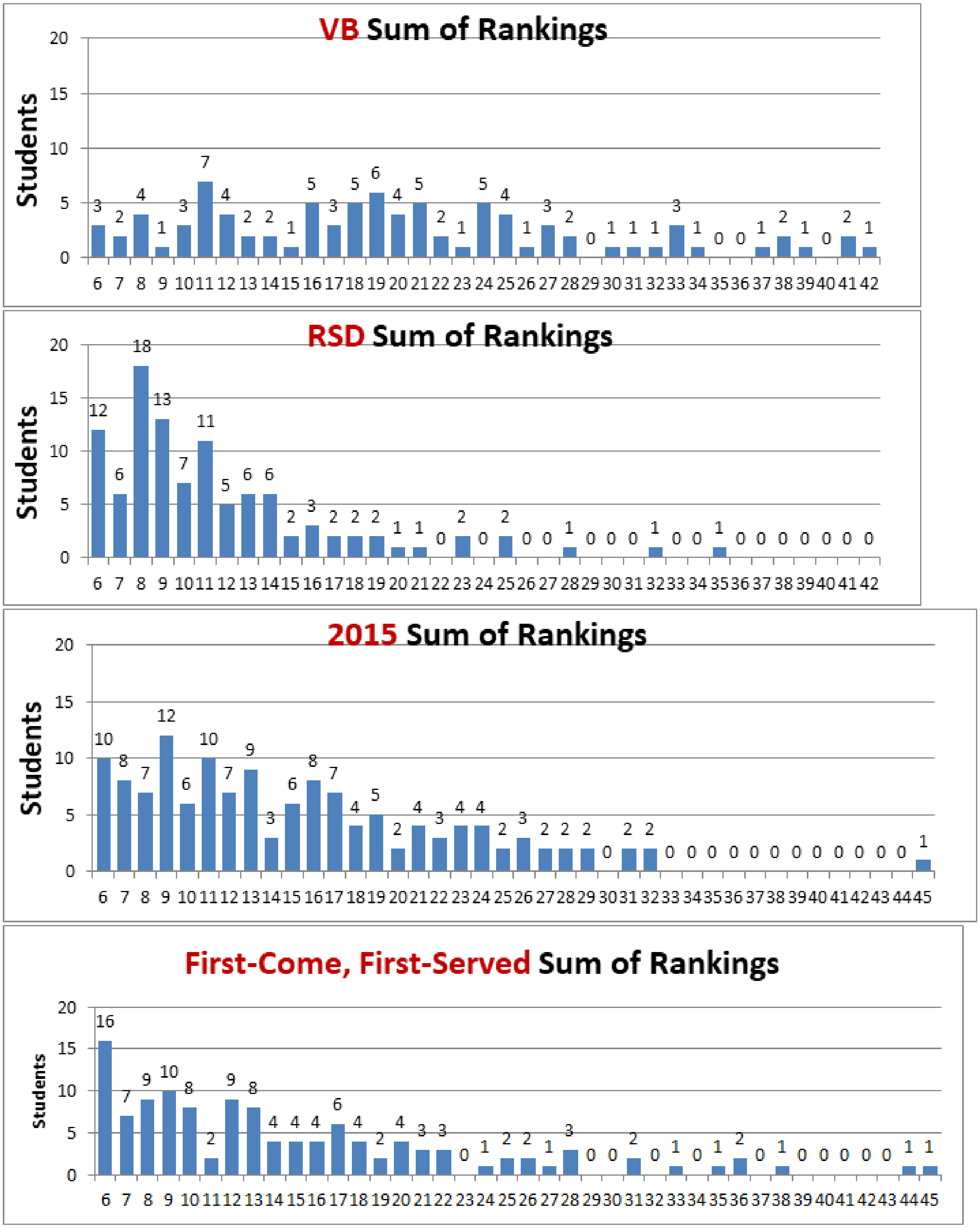
Table 18: Sum of Rankings Statistics

	RSD	VB	2015	FCFS	FCFS Penalty
Maximum	35	42	32	45	72
Minimum	6	6	6	6	6
Average	11.60	19.88	14.90	14.92	18.71
Median	10	19	13	13	13
Std Dev	5.58	9.07	6.83	8.56	14.84

[98] The VB program no longer is functional, even when running on this identical data. Additionally, the VB program is not repeatable: due to the random nature of the Monte Carlo algorithm and the manner in which the VB algorithm was implemented: no two runs ever yielded the same results.

[99] After student registrations were completed, there were nearly a dozen students (parents) who subsequently indicated that they were no longer available in a few time slots, but had suddenly become available during other time slots. The correlation was 1.0 that the new availability matched the student's highly ranked Events, and the reduced availability corresponded with the student's low ranked Events. Apparently, some coaches or other people "in the know" may have leaked the Event schedule time slots to a select few families…

Figure 14: Event Assignment Misery Index



4.1.4 Stable Marriage Outcome

While Gale-Shapley[1] is a two-sided matching mechanism[100] and Event Assignment is a one-sided matching mechanism, the concept of stability in the match result applies to both mechanisms. With Gale-Shapley's stable marriage, all four parties involved in a pair of marriages must agree to any partner swap, otherwise the existing arrangement is considered to be stable when one or more of the parties would disagree to a proposed swap. With Event Assignment, there are only two parties that would need to agree to a swap.

Applying a modified Gale-Shapley definition of stability to Event Assignment solutions:

> no student, A, should be willing to trade one of their assigned Events, X, for an Event, Y, from any student, B, where B values X more than Y and A values Y more than X.

In any Event Assignment solution, if student A and B are willing to trade Events, then the solution is not stable. If no two students are willing to trade Events, then the solution is stable.

Note that a "stable marriage" only applies when the objects on both sides of the match have their own independent preferences for objects on the other side of the match. In the case of SO, students have preferences for Events, but Events do not have preferences for students. For Event assignment, the concept of a Gale-Shapley Stable Marriage doesn't strictly apply. It is possible, however, to run a modified Stable Marriage check on the resulting Event Assignment and see whether the results are compliant or not with the definition of a modified Stable Marriage. Stability would preempt any student wanting to trade Events with any other student on a bilateral basis.

In this last year of the project, code was added to the RSD to check and verify the Stable Marriage rules after every EA run. The check consists of the following logic:

```
For ( studentI = 0; studentI < maxStudent; studentI++ )
    For ( EventI = 0; EventI < maxEventsForStudentI; EventI++ )
        For ( studentJ = studentI + 1; student < maxStudent; studentJ++ )
            For ( EventJ = 0; EventJ < maxEventsForStudentJ; EventJ++ )
                If ( studentI prefers EventJ >  EventI  AND                    (20)     .
                     studentJ prefers EventI > EventJ )
                        If ( studentI is allowed to take EventJ  AND
                             studentJ is allowed to take EventI )
                                NOT STABLE MARRIAGE;
```

Event Assignment checks out as always having (modified) Stable Marriage results. The check does flag Event assignments for pairs of students where, given no constraints on selection[101], the students would be willing to trade. However, in each case, there is a hard constraint that prohibits the trade from occurring, whether it is due to a test, schedule, or availability conflict. The check does not need to look at capacities since the students would be swapping Events for a net-zero change in enrollment.

---

[100] A two-sided matching mechanism is where both sets of agents/items being matched have preferences for the other. The classic example from Gale-Shapley is a set of men and a set of women who each have an ordinal preference ranking for members of the other set. A more relevant example would be students have preferences for which Events they take and the instructors for those Events have preferences for which students are in their class.

[101] No constraints means to ignore student availability, teaching and testing conflicts.

This result is as would be expected since in each Round of assignment, each student is assigned to the highest preference ranked Event that they are allowed to get. This means that at no point is a student assigned an Event that is lower on their preference list than the best available Event they prefer.[102] And, because there is no reciprocal preference by Events for students, there cannot be a situation where two Events would want to trade students.

4.1.5 Pareto Optimality

"In such a setting where the choices of agents are in general conflicting, economists regard Pareto optimality (or Pareto efficiency) as a basic, fundamental criterion to be satisfied by an allocation. This concept guarantees that no agent can be made better off without another agent becoming worse off. A popular and very intuitive approach to finding Pareto optimal matchings is represented by the class of *sequential allocation mechanisms*[103]."[8] (2)

As described in section 4.1.4, a check can be done on the results of the Event Assignment to see if any two parties are willing to trade any Events with each other. The two parties are only willing to trade if at least one of them sees an improvement in their MI. Since ordinal rankings are used to assess the value of Events for each student, any trade will result in both parties experiencing a change in their MI. This is because no two Events can have the same ordinal rank value for a single student: no student is allowed to rank Event X the same as Event Y. Trading X for Y or Y for X results in a change in MI for the students involved in the trade. Thus, both parties would have to experience an *improvement* in their MI if a trade were to be made. If the stability check does not find any two-party trade(s) that can be made, the assignment result is considered Pareto Optimal.[104]

An additional question can be asked about whether or not the assignment result is an "equal distribution of the available wealth".[4] (222) The answer is no, it is not equal since students have different MI values, and the range of values can be rather wide. Due to the testing constraints and the differences in availability and personal preferences of individual students, the results are almost always guaranteed to be unequal between students.

Event Assignment is an initial distribution of 'wealth' rather than a redistribution of 'wealth'. That is, no student starts with an endowment of Events that are potentially redistributed across other students by Event Assignment. It might be possible to make some adjustments to the algorithm or the process that could yield a Pareto Optimal result that is also more equal in distribution than other Pareto Optimal results. It would be important to arrive at this more equal distribution without making team selection more difficult. For example, minimizing the teaching conflicts or rearranging the schedule such that Events that are highly favored by students occur in timeslots where most of those students are available, and then reviewing the statistical results for the student pool could be informative in making those modifications. The result in the overall MI of the student pool might be improved and/or the range of MI for the students might

[102] It should also be noted that if two students had tried to game the system (by swapping preferences) and lost, and subsequently modified their preferences back to their true preferences, it is possible that these two students might want to trade. However, it is unknowable whether this could happen since we cannot collect their true preferences after the fact.

[103] Sequential allocation mechanisms include all forms of serial dictators.

[104] Strictly speaking, in a multi-player environment such as Event Assignment, Pareto Optimality would be achieved if no group of *N* students in the pool would be willing to trade one or more of their Events. Checking for this condition would essentially result in re-solving the assignment problem through an exhaustive search which quickly becomes computationally unreasonable as it morphs into an NP-complete problem.

be narrowed. Arguably this would mean the result is a more equal distribution of wealth amongst the students. However, there would still be individual students who would be worse off in the more equal distribution than in a less equal distribution. Thus, in comparing one possible result to another, those results would not be Pareto Optimal with regard to each other.

With the confirmed stability of the Event Assignment and its Pareto Optimality, there is no need to have a secondary market for students to trade Events due to inefficiencies in the algorithm.[105] In addition, once Events start being taught, it does not make sense to trade students between Events because a new student moving into an Event is behind the other students in that Event. This ‘trade’ results in extra work for the Event Coach and the student in trying to get caught up with the other students.[106]

4.1.6 Time and Labor Savings

A very great advantage to the automated version of Event Assignment with the SOTM website is the tremendous amount of time and labor saved versus the prior methodologies of Team Selection and Event Assignment using spreadsheets, manual team selection, manual timetabling, and the VB script for Event Assignment of each team. While there are no official records of the amount of time and labor that went in to this VB script process, the calendar time between the close of student registration and the announcement of teams and Events for the students varied from 6 to 9 days. Anecdotally, one of coordinators from 2014 reported that they spent 4 to 8 hours each day for 7 days iterating on moving students between teams, contacting families to see if they could modify their availability, and re-running the VB[107] script to get 6 full teams.

When run manually, the Event Assignment with the RSD would take between 1 and 4 hours total time per school. Schools that ran the Event Assignment on the full pool of students without the need to pre-select teams would take between 1 and 2 hours to complete the process. Schools that needed to pre-select teams and perform Event Assignment within a team would require between 2 and 4 hours of work to complete the process.

With Event Assignment integrated into the SOTM website, Event Assignment currently can be completed between 5 minutes and 1 hour for the full pool of students, or between 30 minutes and 3 hours to pre-select[108] teams and assign Events within each team.

---

[105] In the second year of this project, we did allow students to submit requests for Event trades via email in the first week following Event Assignment. Of the 135 students in the SOP, only one pair submitted a request for a trade. Both students had to agree to the trade via separate emails from their email accounts. Based on the students’ preferences, student A was getting a much higher preference Event, and student B a much lower preference Event—thus violating Pareto optimality. Within an hour of the trade request, student B’s parent sent an email nixing the trade. It appears that student A bullied or otherwise intimidated student B into agreeing to the trade.

[106] In a novel experiment in the SOP the two years before this project, the Head Coaches decided to swap students between Events based on the Event Coaches’ evaluations of their students after 3 weeks of meetings. The lower-rated students in 22 Events were swapped around, resulting in as many as 10 of 12 students in an Event being moved to other Events. The 22 Event coaches had to re-teach 3 weeks of material, and compress their schedules by those 3 weeks in order to cover all of their respective material in time for the Regional Competition. (Note that the 23rd Event had no students swapped, nor were any of the Head Coaches’ kids’ Events swapped.)

[107] It was also reported that the run duration of the VB algorithm was unpredictable and prone to frequent crashes, where the crash state was indistinguishable from the program continuing to run and just not yet completed.

[108] Generally, the elementary schools will pre-select teams at the start of the year and middle schools will form teams mid-year. This is due to the age and maturity level of the students. Curiously, at high schools where the SOP is run by a student committee, the teams are pre-selected at the start of the year based on prior year performance—and freshman are prohibited from being on “Team 1” regardless of their prior year SO performance.

To further assist head coaches with Event Assignment, statistical information on the Event Assignment results are displayed on the GUI. This includes information on the overall MI of the pool, number of students in each Event, how many students were assigned to their top selection of Events, and so on. The Head Coaches can also scroll through and sort the results both by a student list and by Event rosters. This information enables the Head Coaches to make a more informed decision concerning which Event Assignment parameters yield the best results for their SOP.

4.2 Team Selection

When it comes to parents and their kids, having their kid chosen (or not chosen) for a higher-ranked team can be a very emotionally charged occurrence. The unfortunate tradition with manual selection is that those students being selected for the top teams are usually chosen more because the Head Coach knows the student (or the family) than because the student is a better performer. Often times, the assessment of performance is highly subjective and open to interpretation. Thus, when teams are selected by humans as a result of their subjective performance assessments, the selected team is often met with parental discord and cries of favoritism.

4.2.1 Binary Search

The first version of the Team Selection algorithm was written as a greedy search algorithm that started from random points and explored and tested as many combinations and permutations of students and Events as it could in the allotted run-time to make a team from the pool of students. The program was straightforward to write: essentially consisting of a series of inner and outer loops, checks to make sure the combination created for the team adhered to the rules and constraints for a team, and then comparing the new team score against the current best team score. Each time a higher scoring team was discovered, it became the new candidate team. In addition, the program was straightforward to enhance to use parallel processing using an N-Greedy algorithm: determine the number of CPUs to be used and clone the process to the other CPUs, giving each one a random start point from which to explore. While Greedy algorithms may get stuck in a local optimum, the N-Greedy enhancement explores a wider range of the solution space, making it less likely that the best solution found is a relatively low local optimum.

This was not too onerous of a computational task, as there was access to a farm of some 2,000 CPUs on which to run the Team Selection over long weekends (typically Thanksgiving for the Regional teams and President's Day Weekend for the State team). In the first year, the algorithm ran for 96 hours on 1,000 CPUs to find the State team from a pool of over 130 students. It was not known if that was the optimal team since the algorithm would have had to finish checking every combination and permutation before it could declare the solution optimal, and the algorithm did not run to completion in the allotted time window.

For the Regional teams, it was not possible to run the 6 teams in parallel, since team 2 and all subsequent teams depended on the composition of team 1 and all prior teams because each team was drawn from the student pool without replacement. Thus a lower-tier team could not be formed until the team prior to it was removed from the student pool. The algorithm used to select the Regional teams using *N* CPUs was a variant on an N-Greedy algorithm:

1. Select the *N* top sub-optimal teams for Team 1

2. For each solution of Team 1, exclude the students and select the top *N* sub-optimal teams for Team 2
3. Sort the list and take the top *N* solutions from the $N_1 * N_2$
4. Repeat step 2 and 3 for the remaining teams

While the algorithm did not necessarily find the optimal top teams, they were close enough and the teams were tiered from 1 to 6 in their potential performance.

4.2.2 Integer Optimization

Binary Search Team Selection took too much time and required too many privately-owned resources to be a process that could be automated to be run on-demand by head coaches from a small web site. Because Team Selection is a combinatorial optimization with discrete decision variables and constraints that can be formulated with linear inequalities, it easily lends itself to be converted to an Integer LP (IP). Compared to the brute-force binary search algorithm, good solutions should be able to be found much faster, and the IP solver will report if the solution found is optimal or not—and if not, it also provides the theoretical upper bound of the optimal solution.

I formulated the IP described in section 3.6 and implemented the problem formulation as a C program that can run on the SOTM web server. The problem file can then be solved locally on the web server[109] with a single CPU or exported to the NEOS server[110] for faster solutions.

The speedup from this conversion was tremendous—much, much larger than initially expected. Locally, a first solution to a team can be found in anywhere from under one minute to three minutes, and the optimal solution confirmed, worst case, within 8 hours. For Regional teams, each subsequent team is found exponentially quicker than the prior team. Thus, for 6 teams solved to optimal takes less than 24 hours with a single CPU, or all 6 teams can be found in less than 30 minutes if 100% optimality is not required—for example, accepting 90% of the theoretical optimum for the first two or three teams. Again, contrast this speed with the original Team Selection (Greedy) implementation which took 96 hours on 1,000 CPUs to get close-to-optimal teams. Note also that the solution time for the Greedy implementation to find 1 team or 6 teams is the same due to the manner in which it was coded.

Table 19: Team Selection Solution Times

Solver	CPUs	Solution Time	Pool Size	Optimal?
Manual	Humans	~7 days	90	No
Greedy Search	Up to 1,000	96 hours	142	Not confirmed
XA	1	8 hours	~120	Within 3%
NEOS/Gurobi	Variable, 1 to several	0.03 seconds	~120	Yes

The NEOS server provided even greater speedup, as the actual computational time to solve to optimal for a single team was as low as 0.03 seconds! The transmission time of the problem file, waiting in a queue, and solution return via email is actually the slowest part of the NEOS

[109] Using the XA LP/IP/MIP solver donated by Sunset Software in Pasadena, CA.
[110] "NEOS is a free, internet-based service for solving numerical optimization problems. Hosted by the Wisconsin Institute for Discovery at the University of Wisconsin in Madison". https://neos-server.org/neos/

process, yielding a total round-trip time from problem to solution of 2-3 minutes per team. Total wall clock time for generating 6 sequential 100% optimal teams takes less than 18 minutes.

As this is an IP and the formulation is NP-complete, these solution times can be greatly affected by the number of students in the student pool. The solution times in Table 19: are representative for student pools of the stated pool sizes.

4.2.3 Scoring Issues

There have been a number of issues that have been encountered over the years with regard to scoring student performance in the grade book. There have also been several different solutions tried to correct or minimize these issues, from trying to educate coaches about grading trends and the need to have objective scores for the students to implementing mathematical bounds on the Event coaches' subjective behaviors. Given that the number of issues each year stayed constant using only the educational solutions and given that the effect of the issues has been diluted when using the mathematical bounds, it's pretty clear that continuing the use of the mathematical bounds is the most effective and consistent way to ameliorate the scoring issues.

The primary scoring issues experienced have been:

- Inaccurate Scoring – where a coach's assessment of relative student ability does not match the results from 3rd party assessments
- Non-discriminating scores – where the coach scores all students the same or no scores are recorded, which is effectively the same issue)
- Parental Bias – where the coach inaccurately scores their own child higher than other students in the Event
- Less Effective Instruction – where the coach does not prepare material in advance, doesn't know how to work with the age range of their students, does not understand the material, teaches the wrong material, or signed up to be a coach to get the assignment incentives for their own child but with no apparent desire to teach
- Free Riders – Students who are paired with higher performing students in most or all of their Events who consequently share in the performance of the partner, artificially raising their own scores[111]

The 2020 season was a representative year for coaches when it came to non-discriminating scoring. For the 23 Events, assignment scoring entries are shown in Table 20:. There were 5 Events where the coach entered 2 or fewer assignment grades—and in all 5 of these Events, all the students in each Event were given the same grade (100%), thus providing no information with which to discriminate between the students' abilities.

Table 21: shows the number of Events where the coaches scored all students in their Event with perfect Citizenship (4 Events), perfect Attendance (5 Events), or both Citizenship and Attendance as perfect (2 Events). There were 11 Events where the Event coaches threw away an opportunity to discriminate between students based on behavior and reliability. Only in half (12) of the Events did the Event coaches track and discriminate on based Attendance and Citizenship.

[111] This problem was a much more significant problem under the old SO regime, where students were assigned to teams at the start of the year and there was no tracking of individual student performance. The head coaches would specifically pair their own kids with the highest performing kids to improve the chances of their kid advancing to the state team. There were also Free Riders that would get this 'advantage' by random chance of assignment as well.

Table 20: Grade book Event Assignment Scoring

# of Assignments	# of Events	Discriminating
0	1	No (1)
1	2	No (2)
2	2	No (2)
3 to 5	6	Yes (5) No (1)
6 to 10	8	Yes (7) No (1)
> 10	4	Yes (4)

Table 21: Events with Perfect Attendance and Citizenship

	Perfect Citizenship	Perfect Attendance	Both Perfect	Tracked Both
# Events	4	5	2	12

Taking a more global look at the Event grade books and the performance of students in those Events at competitions and the Regional, a fairly comprehensive and dense table of information can be made and is shown in Table 22:. This table illustrates the grading issues experienced in SO with volunteer (non-professional) teachers. The table consists of these columns of data:

- Event – the acronym for an Event
- Light Orange Columns[112] include data on the veracity or accuracy of the Event coaches' relative ranking of student ability within their Event versus the ability rankings from the competitions. Students were assigned ordinal ranks based on their grade book scores, ordinal ranks based on their invitational performance, and ordinal ranks based on their Regional scores. These ordinal ranks for each student were summed up and then sorted yielding a final ordinal ranking of the students. An absolute delta between these final ordinal ranks and the coach's grade book rank for the students was calculated and summed over the students within an Event. The average deviation was calculated for each Event, and then all the Events sorted by this average deviation, and that ordering of Events is presented in the table.
    - Total – the total absolute deviation in student rankings for the Event
    - Average – total/number of students in Event = average rank deviation for the Event
    - Max – the greatest deviation between ranks for a student in the Event
- Slate Blue Columns include the data on the grades entered by a coach to get an indication of how well the coach was able to discriminate between students on mastery of their Event.
    - Delta Grade = Highest grade – lowest grade in the Event, out of a maximum range from 0 to 100 points.

[112] The column colors refer to the color of the column header. The colors of the various cells within the columns are described later.

  - # Grades – the number of assignments graded in the Event
  - Cit – Citizenship scores recorded in the Event. P – perfect score for all students. T – tracked, meaning at least 4% of the scores were not marked as perfect.
  - Att – Attendance scores recorded in the Event. P & T as with the Cit column.
  - Disc – Discriminating Scores, were the scores sufficiently complete and diverse to have a minimal ability to separate student performance. Y – yes, sufficient. N – no, not sufficient. N was assigned to all Events with one or both of these characteristics: 2 or fewer assignments or Delta Grade < 20.
- Gray Columns include the data on ordinal ranks of the student teams in the Events at the Regional competition to get an indication of how effective the coach was in teaching mastery of the Event to their students.
  - Best – highest ordinal rank achieved by a team from this school in the Event
  - Worst – lowest ordinal rank of a team from this school in the Event
  - Avg – the average ordinal rank of all teams from this school in the Event
  - Top 3 – the sum of the ordinal ranks of the top 3 teams from this school in the Event
- CK Column – the Coach's Kid column shows whether or not the Event coach's kid was in the Event and whether or not that student's relative competition performance approximated the grade book score awarded to the student. k – indicates that the coach's kid was in the Event and that the performances matched within 2 ranks. B – indicates that the coach's kid was in the Event but that the performances did not match.

4.2.3.1 Inaccurate Scoring

From the light orange columns in Table 22:, it can be seen that the average deviation between a coach's ranking of students and the competition rankings ranges from 0.55 to 3.78, where the lower values show a close match between the two rankings and the higher values show wider deviations. Those Events and coaches with average deviations of 1 or lower mean that their students were ranked by the coach within 1 of their final performance rankings at Regional—meaning very accurate rankings. Those Events and coaches with deviations of more than 2 mean that the coaches did not do a very accurate job evaluating student performance for their Event—especially considering that there were no more than 12 students in an Event, and as few as 7 students. Thus, being off by 3 is effectively no better than ordering the students by random draw within the Event.

Based on this representative data, the quality of the job that Event coaches did in assessing the relative performance of their students breaks down as:

- 6 Event coaches (rows with green background) did exemplary work
- 7 Event coaches (rows with red background) did poor (nearly random) work
- 10 Event coaches (rows with white) did acceptable work

Note that the task of assessing students' relative performance in an Event does not necessarily translate into absolute performance of the student teams at the competitions. That is, a high ability of the coach to assess student performance does not mean high student performance—especially not for all teams from the school in the Event. For example, while ANAT had the best assessment performance, ANAT also had the 4th worst *overall* performance by teams with an average team ranking of 18.7 at Regional; similarly, FS had the 2nd worst assessment performance and the 3rd best *overall* performance by teams with an average team ranking of 9.5.

Table 22: Event Performance vs. Grade book Scoring

Event	Total	Average	Max	Delta Grade	# Grades	Cit	Att	Disc	Best	Worst	Avg	Top 3	CK
ANAT	6	0.55	1	64.8	9	P	T	Y	6	32	18.7	27	k
ORN	4	0.57	2	29.5	12	P	T	Y	1	20	11.8	27	k
MET	6	0.67	3	60	10	P	T	Y	1	17	7.3	22	
STAR	6	0.86	2	54.5	8	T	T	Y	4	18	10.3	31	
WQ	8	1.00	3	45.8	12	T	T	Y	1	26	11.3	19	
DD	10	1.00	4	52.4	6	T	T	Y	1	14	9.4	20	
MACH	10	1.11	3	26.6	5	P	T	Y	1	25	12.4	16	k
RS	10	1.11	3	44.4	**2**	T	T	N	4	21	10.8	20	
CODE	14	1.17	2	74.6	15	T	T	Y	3	33	18.2	23	k
WIDI	16	1.33	3	49.7	**3**	T	P	Y	1	29	15.2	20	
MP	16	1.33	4	40	8	T	T	Y	8	27	17.8	42	B
DP	14	1.56	4	95	5	T	T	Y	2	15	10.5	27	
FOS	14	1.56	6	52	5	**P**	**P**	Y	1	17	10.3	24	
PPP*	22	1.83	4	**6**	5	P	**T**	N	9	20	14.0	31	
CB	22	1.83	5	59.4	**3**	T	T	Y	3	24	11.2	12	B
EXPD	28	1.87	8	23.1	6	**P**	**P**	Y	2	20	10.8	20	
DL	26	2.17	7	78.6	17	T	P	Y	2	26	13.2	19	B
CL	24	2.18	6	48	7	T	T	Y	3	36	21.0	32	B
BOOM*	26	2.60	6	60	**1**	**T**	**T**	N	1	32	14.4	16	
ELG*	32	2.67	5	**3**	**1**	**T**	**T**	N	11	31	21.3	48	
MV*	32	2.67	6	**0**	**0**	**T**	**T**	N	26	35	30.2	83	B
FS*	38	3.17	9	**16.1**	10	**T**	P	N	1	17	9.5	12	k
HER*	34	3.78	7	**0**	**2**	**T**	P	N	10	27	17.6	36	B

4.2.3.2 Non-Discriminating Scoring

In the slate blue columns in Table 22:, it can be seen that the Event coaches near the top of the table did a better job discriminating student performance than the coaches near the bottom of the table. The coaches near the top of the table also had more success in accurately assessing their students' performance and, thus, in forecasting their students' performance in their Event at competitions. From in class observations of various coaches over several years, it is clear that coaches who engage the students by using a variety of instructional methods engender higher student performance than those who drone on through 90 minutes of lecture each class session. These instructional methods usually include a variety of lectures, demonstrations, hands-on activities, creating high quality tests, making multiple (and varied) assessments throughout the

SO season, providing subject material at an appropriate level, student presentations, Q&A, and simply being organized[113].

4.2.3.3 Parental/Scoring Bias

Anywhere from 1/3 to 1/2 of parent coaches whose kids are in their Event are noticeably biased toward their own kids in their classes and they score their kids from a little to significantly higher than is accurate relative to other students in the Event. This is seen when the students attend tournaments and their tournament performance does not track their grade book score.

Table 23: Event Performance by Regional Results

Event	Total	Average	Max	Delta G	# Grades	Cit	Att	Disc	Best	Worst	Avg	Top 3	CK
MET	6	0.67	3	60	10	P	T	Y	1	17	7.3	22	
DD	10	1.00	4	52.4	6	T	T	Y	1	14	9.4	20	
FS*	38	3.17	9	**16.1**	10	**T**	P	N	1	17	9.5	12	k
FOS	14	1.56	6	52	5	**P**	**P**	Y	1	17	10.3	24	
STAR	6	0.86	2	54.5	8	T	T	Y	4	18	10.3	31	
DP	14	1.56	4	95	5	T	T	Y	2	15	10.5	27	
EXPD	28	1.87	8	23.1	6	**P**	**P**	Y	2	20	10.8	20	
RS	10	1.11	3	44.4	**2**	T	T	N	4	21	10.8	20	
CB	22	1.83	5	59.4	**3**	T	T	Y	3	24	11.2	12	B
WQ	8	1.00	3	45.8	12	T	T	Y	1	26	11.3	19	
ORN	4	0.57	2	29.5	12	P	T	Y	1	20	11.8	27	k
MACH	10	1.11	3	26.6	5	P	T	Y	1	25	12.4	16	k
DL	26	2.17	7	78.6	17	T	P	Y	2	26	13.2	19	B
PPP*	22	1.83	4	**6**	5	P	**T**	N	9	20	14.0	31	
BOOM*	26	2.60	6	60	**1**	**T**	**T**	N	1	32	14.4	16	
WIDI	16	1.33	3	49.7	**3**	T	P	Y	1	29	15.2	20	
HER*	34	3.78	7	**0**	**2**	**T**	P	N	10	27	17.6	36	B
MP	16	1.33	4	40	8	T	T	Y	8	27	17.8	42	B
CODE	14	1.17	2	74.6	15	T	T	Y	3	33	18.2	23	k
ANAT	6	0.55	1	64.8	9	P	T	Y	6	32	18.7	27	k
CL	24	2.18	6	48	7	T	T	Y	3	36	21.0	32	B
ELG*	32	2.67	5	**3**	**1**	**T**	**T**	N	11	31	21.3	48	
MV*	32	2.67	6	**0**	**0**	**T**	**T**	N	26	35	30.2	83	B

There were 11 parent coaches in 2020 in one school who had their own child in their Event, 6 graded their kids in the top 2 in their class, but competition performance ranked them in the middle to bottom of their Events. The remaining 5 graded their kids such that their competition performance matched their relative grade book scores. Since there were 4 competitions in which the students competed, the consistent deviation of scores from those competitions gives weight to

[113] These factors were noted over the years from observing the various coaches while teaching their Events, canvassing coaches about what techniques worked or didn't work in conveying material to the students, as well as through student surveys reviewing their Event coaches at the end of the SO season.

the idea that 6 coaches favored their own children with regard to grade book scoring. From Table 22: in the CK (Coaches' Kid) column, there is also a large proportion of those coaches who tend to favor their own children who are also not particularly skilled at assessing the performance of the students in their Events as their record shows that their assessments in the grade book don't track well with student performance in the competitions.

In one particularly egregious case of parental bias, a head coach accurately entered all of the Invitational and Regional Competition scores for 22 of the 23 Events. In the 23rd Event, which included the head coach's kid, the head coach entered only two students' Invitational scores (out of 6 total Invitational scores that should have been entered, meaning those other students received zero's and significantly lowered the scores of those students) and did not enter *any* of the Regional scores. The result was that the head coach's kid was ranked 2nd in the Event with a composite score of 86% and the 3rd ranked student in the Event had a score of 42% (with the rest of the students scattered between 42% and 26%). Algorithmically, the head coach's kid was "in" for the State Team that year because the head coach had engineered such a large gap between the scores of his kid and the next best kid in the Event. After entering the correct scores for the Event and then re-running the State Team selection, the head coach's kid did not get selected for the State Team[114].

4.2.3.4 Less Effective Instruction

Shown in Table 23: (which is the same data as shown in Table 22: just sorted by the Gray column Avg [then Top 3, then Best]) is the order of Events by their overall performance in the Regional competition, with the best performing Events at the top of the table, and the lowest performing at the bottom of the table. While the correlation is not 1.0, it is apparent that the better performing Events tend to be those Events with more accurate assessments of the students and grade book scoring that highlights student differentiation.

There does not seem to be much that can be done to 'fix' ineffective instruction beyond making a coaching change. Bookkeeping and mathematical adjustments to the data can't change the classroom environment or the amount of learning that students undergo. Usually, by the time an Event is identified as having an ineffective coach, the season is quite far along and there is not a ready pool of volunteers itching to dive in and take over an Event. This makes swapping in a new coach rather difficult. After the Regional competition, many of the less effective Event coaches resign, and it becomes easier to recruit one of the parents of the two students in the now-coachless Event to take over as the coach. If one of them does not step up, then the students would be in a difficult situation where they would have to work on their own. The parents at these schools are very unlikely to let that happen.

Coaching changes can result in significantly improved student performance from Regional to State. Coaching changes and the resulting change in student performance was not tracked in the data, so there is no hard data to support this, only anecdotal instances where students improved from finishing below 20th at Regional to top 5 at State[115]. The main instances where there was no improvement or a significant drop in performance were in some of the build Events, where a

---

[114] I did not notice the data discrepancy until more than a year after that SO season was over. Curiously, the Event coach did not appear to notice the data discrepancy at the time either, and it begs the question of whether the Event coach was simply not paying attention or possibly complicit in the 'oversight'.

[115] Examples of this include a lab Event where the students moved from 20th at Regional to 1st at State, a robotics Event where the students moved from 37th to 9th at State, and WIDI where students moved from the high 20's to 1st at State.

parent of one of the students in the build Event would attempt to re-engineer and 'improve' the device in the last few days before the State competition. This most frequently resulted in complete failure or a significant drop in performance[116].

4.2.3.5 Impact on Team Selection

The discussion around the grade book and scoring tendencies of Event coaches is significant because of the effect that scoring has on Team Selection. The objection function for Team Selection is based on the grade book scores of the students in their Events for Regional Team selection and both the grade book and competition scores for State Team selection. When that data is reliable and accurate, then the selected teams are likely to be the optimally tiered teams. When that data is unreliable, inaccurate, or random, then the selected teams cannot be the optimally tiered teams in an absolute sense. Thus, when there are between 5 and 8 Events each year where the grade book scoring is at best inaccurate and at worst scored without any differentiation amongst student abilities[117], there is a significant amount of damage that is caused to the strength of the selected teams: "Garbage In, Garbage Out". Due to the combinatorial nature of team selection, the garbage data in one or more Events pollutes the better data in other Events.

Conversely, given the combinatorial nature of the Team Selection algorithm where students have multiple Events, there is some dilution of the random effect of the poorly-scored Events. For students in a poorly-graded Event, the performance of those students in their *other* Events with *accurate* grading will partially counteract bad data from the poorly-graded Event.[118] Unfortunately, for the few students who have most or all of their Events poorly-graded, those students are essentially being assigned random numbers for performance scores and their fate in Team Selection is completely dissociated from their ability.

4.2.4 Scoring Mitigation

Because the coaches in this environment are all volunteers and generally participate in SO only for a handful of years while their kids are in SO, there is not a lot of opportunity, time, or support infrastructure available to correct or improve the teaching and scoring habits of the coaches with the more obvious deficiencies. And it is not particularly clear that these adults, who are often

---

[116] An example of this is where the night before the State competition, a parent sprayed lubricant on a plastic roller coaster track to 'speed up' the roll rate of the ball. The lubricant reacted with the track overnight and warped and wrinkled the surface to the point that the ball fell out of the track.

[117] In the representative data, there are two Events where all students were scored the same: MV had no assignments at all, and HER had two assignments where all students were scored at 100%, thus neither Event had any differentiation. Due to the structure of the grade book, the available differentiation for those two Events was supplied entirely by Citizenship and Attendance. While students who attend classes and are better behaved in class tend to learn more in the class, Citizenship and Attendance are still far from an ideal measure of mastery—but they are still better than random rankings.

[118] Typically, however well a student performs in one SO Event, they will tend to do similarly well in their other Events. Thus, the counteraction brought about through the combinatorial interdependence for a student will tend to bring their score in a poorly-graded Event closer in-line to their performance in their other (hopefully accurately-graded) Events. This effect can work both ways, effectively raising a student's performance score in the poorly-graded Event or dropping it depending on their other Events.

well-established in their careers, would be open to constructive criticism or willing to spend time on improving their teaching performance.[119]

In order to mitigate these scoring issues, five mechanisms have evolved:

1. Dilution
2. Normalization
3. Objective Assessments
4. Event Weighting
5. Coach Replacement

The Dilution, Normalization, and Event Weighting mitigation all are implemented within the software. Objective Assessments are in the hands of 3rd party individuals affiliated with other school's SOPs or Regional SO. Coach Replacement is the only mitigation element entirely in the hands of the head coach.

Even though the head coach has the ability to set the Normalization and Event Weighting parameters in the software, there is a locking mechanism in the software that requires the parameters be set and locked before the head coach may use the results from Event Assignment and Team Selection. The head coach is also expected to report the parameter setting to the stakeholders in advance of their use. So, even the head coach is greatly limited in their ability to influence those software results because at the point in time where the (final) data is known and there would be opportunity to 'play' with the parameters to generate different outcomes, those parameters have been locked.

Each of these mechanisms for scoring mitigation is discussed in the following sections.

4.2.4.1 Dilution

Dilution handles this scoring bias (for students with 3 or more Events[120], where these are most likely the students who are on higher-tiered teams that have a chance at setting the school trophy). The dilution occurs naturally due to the combinatorial nature of Team Selection: all of a student's Event scores contribute to the team score. A student who has a high bias score in one Event will have the effect of that high score diluted by the student's other Event scores: if a student has 3 Events, the high-bias score will be effectively diluted by 2/3's and with 4 Events, the dilution is ¾. A student with 2 Events has the bias advantage diluted by ½, and a student with a single Event will not undergo any dilution.

Also, because students earn individual scores in the grade book and only share scores from competitions, the grade book scores provide differentiation from the competition scores. Even though the grade book may not be weighted as heavily as the competition scores, the grade book still provides the potential for up to its percentage weight in the overall rubric points for differentiating between students. This differentiation helps inhibit Free-Riders from being selected for the higher-tiered teams.

119 My personal experience has been that the less effective an adult is in conveying information and the less able they are to accurately assess student performance, the less open the adult is to any sort of criticism or offers of help or suggestions on how to improve the learning and performance of their students. The good teachers, on the other hand, are always talking with other coaches about new ideas and approaches and trading commentary on techniques that work and those that don't work.

120 Students with 2 or fewer Events typically do not get selected for the higher-tiered teams unless there is a plethora of 4-Event students and the 2-Event student scores high in both their Events.

The weighting of the competition scores also serves to dilute any bias-grade book-scored advantage (see section 4.2.4.3).

If the competitions are weighted at 80% in the overall rubric and the grade book is weighted at 20%, the grade book advantage is diluted 4:1 by the competition scores and diluted by a further 2/3's for a student with 3 Events. While a parent coach may think that they are subtly providing their own kid with a slight advantage for team selection, that impact is actually much smaller than they might believe.

4.2.4.2 Scoring Normalization

In Table 22:, the Grade book scores are entered by 23 very different people (the Event coaches), most with little or no training or experience grading middle school students. In addition, the Event coaches vary rather widely in their ability to teach, and in both their ability and willingness to evaluate student performance objectively. Hence, the Grade book scores entered by Event coaches are best used for determining the *relative* performance of students within an Event.

Similarly, the absolute scores in one Event's grade book cannot be compared to another Event's grade book scores. This is because one Event may have all their students scored within a range of 10 points while another Event may have scores ranging over 90 points, and one Event may have all of their students centered on a score of 95 while another Event may have their students distributed around a score of 50. In an absolute sense with regard to the Team Selection objective function, an Event with high average scores would have more weight on the Team Selection than an Event with low average scores. Therefore, grade book input is normalized to extend over the same range for all Events, so that no one Event carries more weight than any other Event during the Regional team selection.

The formula used for normalizing the grade book is Min-Max Feature Scaling:

$$X' = a + \frac{(X - X_{min}) * (b - a)}{X_{max} - X_{min}} \qquad (21) \quad .$$

Where

X – is the value being transformed

X' – is the transformed value

Xmin – is the minimum value in the source (untransformed) data

Xmax – is the maximum value in the source data

b – is the maximum value of the transformed data range

a – is the minimum value of the transformed data range

4.2.4.3 3rd Party (Objective) Assessment

To further dilute and minimize the effect of parental scoring bias and inaccurate scoring by some of the coaches, more 3rd party assessments were added and their weights increased in the overall rubric[121] over time as it became apparent that scoring was and continued to be an issue. For the

[121] An example of the overall rubric is shown in Figure 4:, where it can be seen that the Coach's grade book score counts as 30% of the Regional Team selection weight, and the mini-comp tests count for 60% of the weight. For

biased and inaccurate scores, the 3rd party scores would tend to bring overall scores for those Events back in-line with reality; for those Events which were reasonably accurately scored, the 3rd party scores wouldn't significantly impact the overall assessments. In either case, the 3rd party assessments would lend confirmation to how accurately the students were being scored, point to those Events where there were deficiencies in teaching and learning, and open opportunities for remedial action prior to the Regional and State competitions.

These independent 3rd party tests come primarily from invitational tournaments, SO Regional competitions, and test exchanges with other schools. The invitational tournaments use tests created by coaches from the various schools attending the tournament, where each school typically writes and runs the test for one Event (thus providing 22 independent Event tests). The Regional competitions use tests created and scored by people who are not affiliated with any local schools but the testers are affiliated with SO and know the syllabi for the current year's Events. Test exchanges with other schools occur when two or more schools write tests for all the Events and then swap those tests with each other. While the test exchange tests are more variable in quality than the invitational and Regional tests, they still provide a testing perspective that is independent from the Event coach at the school.

Like the Event coaches' grade book scores, the test exchange tests yield scores that reflect student performance *relative* to the other students within the same school in the Event since the scores can't be compared to the scores that students from other schools achieved on those tests. Therefore, these scores should be normalized like the Event grade book scores[122].

Unlike the Event coaches' grade book scores, Invitational and Regional competition results provide a comparison between a school's students and students in other schools across a full range of very well-prepared students to poorly-prepared students. This provides an absolute measure of where students are in the entire competitive field. Hence, competition scores need not be normalized and can be used as-is. This is similar to how SAT and other standardized test scores provide a more reliable measure of a student's specific mastery or aptitude of a subject than do their classroom grades[123].[9]

Essentially, grade book scores from Event coaches can be subjective and of uncertain or variable veracity, but 3rd party tests are about as objective measures of student performance as can be reasonably attained.

State, the influence of the grade book score decreases to 9.4% while the invitational (18.8%) and Regional (50%) 3rd party test results count for a combined 68.8% of the scoring weight.

[122] It should also be noted that the test exchange tests are scored by the Event coach (not the author of the test) which means that the test scores are potentially open to manipulation and not truly a 3rd party independent measure of student performance.

[123] There are probably more papers on the usefulness of standardized testing than there are on assignment problems, with many arguing their evidence shows grades are better than standardized tests and just as many others arguing their data shows standardized tests are better than grades. Most of these focus on the predictive power of these two measurements on long-term future success, such as graduating from college and/or future financial success. However, most concede or agree that standardized tests do correlate to aptitude in the subject area better than grades due to different schools awarding different grades for similar short-term performance (apparently as measured by the standardized tests). This application for SO is only concerned about short-term performance, thus trusting student performance based on 3rd party test results more than on grades awarded by untrained, non-professional teachers seems to be a reasonable conclusion based on the data collected during SO.

4.2.4.4 Event Weighting

Early on, head coaches were given the flexibility to set the weightings within Events, but this quickly turns into a guess-and-check enterprise to see if changing the weights moves certain students on or off of a team. Thus, in the last couple of years, the weights were fixed and are no longer available for the head coaches to modify.

4.2.4.5 Coach Replacement

When all other avenues are exhausted, sometimes the only mitigation available is to fire and replace the coach. This clearly cannot be handled by an algorithm or software, and lies entirely in the hands of the head coach. If the students aren't learning, and the coach isn't improving, the last recourse is to ask the volunteer coach not to return to the classroom and recruit a new coach, and hope that the change leads to a better situation for the students.

4.2.5 Intuitive Student Weighting

We had to start somewhere with assigning weights to student performance in team selection.

For Regional Teams, to minimize the Free Rider problem, it was decided to use equal weights for all Events and all students to facilitate the pairing of students of roughly equal ability. Specifically, the scores of student pairs in an Event are calculated as:

$$0.5 * skill_{S_1} + 0.5 * skill_{S_2} = eventScore \qquad (22)$$

Any Event with 3 students would have a total weighting of 1.5 relative to the other Events, with each of the 3 students' performance scores being multiplied by 0.5.

For the State Team, the goal was to maximize team potential with no consideration for Free Riders (under the assumption that the Grade book, dilution, and 3rd party assessments would minimize the probability of Free Riders being selected for the State Team). The weighting consensus reached among the founding adults was:

1. Lab – 50/50. Reasoning: both students needed to have lab technique
2. Study – 60/40. Reasoning: a strong student can bring along a weaker partner
3. Build – 80/20. Reasoning: as per the SO rules, "only one of the student pair need have participated in building the device". By the time of the State competition, the devices have been built and only need to be operated. One 'expert' student can perform any repairs, and an 'assisting' student can quickly be trained on assisting with the operation of the device.

Thus, a novice student can be assigned to a build Event (provided the other student in the pair is an accomplished expert), but it is much more difficult to bring in a novice student to a Study Event and even more difficult to have a novice attempt a Lab Event.

4.2.6 Regressions

To see how reasonable or unreasonable these Student Weightings are, the student composite scores[124] used for Regional Team Selection and State Team Selection were compared to the

[124] Note that the student Composite Scores used for Regional Team Selection have different values than the student Composite Scores used for State Team Selection. For one, the selections occur a few months apart from each other, meaning that the grade book scores will be different due to more Assignments being recorded in the grade book for State. Also, the State Composite Scores will include the results from Regional and Invitational competitions.

paired student performance at the Regional and State competitions. The performance data was divided as shown in Table 24:.

These regression groups were selected for two primary reasons. The first reason being that these are groupings that the National SO uses for Events. The second reason is that the groupings delineate different skill sets or studying styles of students: Build Events involve the engineering and construction of physical devices; Lab Events involve lab techniques and skills; and Study Events involve book-learning and written tests. See also section 4.2.5 for additional discussion on the topic of Event weights.

Table 24: Regression Categories

Event Category	Regional	State
All	Y	Y
Build	Y	Y
Lab	Y	Y
Study	Y	Y

Running regressions on these groups of data yields coefficients that can be viewed as proxies for the weightings of the student pairs' score and their contribution to the Regional or State performance of the pair. That is, according to the Team Selection objective function, student H (higher scoring student) and student L (lower scoring student) contribute their skill in an Event to determine the scoring potential of a Team as:

$$ScorePotential = h_{e1} * skill_{He} + h_{e2} * skill_{Le} \qquad (23)$$

Where,

$h_{e1} >= h_{e2}$ – are the weighting coefficients for the contribution of each student

The performance data for the Regional and State (R+S) Competition performance is provided from the R+S SO organizations only as ordinal place rankings, not as raw test scores as is available from the Invitational Competitions. Thus the R+S results do not provide the intensity of difference in performance between one place rank and another (as is available for the Invitational Competitions). For purposes of scoring and entry of the place rankings into the grade book, the R+S place rankings are converted into percentages according to the formula:

$$score = 100 - (placeRank - 1) * N \qquad (24)$$

Where,

score – is the percentage value to be entered into the grade book for the competition for the student

placeRank – is the ordinal place rank awarded to the student in an Event at the competition

N – is the stepped decrease in score for each place rank[125]

[125] *N* is typically selected to be 3 or 3.333 by the head coaches. This is because place ranks have recently been reported from 1st to 32nd place at Regional competitions and 1st to 30th place at State competitions, and the idea is to award a 1st place team with 100% and a 32nd place team with close to (but not below) 0%.

Data points where a new student takes on an Event[126] or where a student competed as a solo student are not included in the regressions. In either case, the secondary student in that Event is recorded as having a composite score of zero as there is no other data available to assess their ability. In the statistics of Table 25:, there are about 10% of the Regional data points in a year that have a zero for the secondary student score. This is a significant number of data points and they have a significant impact on the regression results, which results in a bias of coefficient weight toward the primary student, as is seen in the bottom 4 rows of Table 25: compared to the top 4 rows of the table. To eliminate this bias, these secondary-zero data points are excluded from the regressions.

In all cases, the model for the regressions was forced through the origin as it would make sense that if a student pair did not show up to compete in an Event, the resultant competition score would be zero, equivalent to last. Similarly, students who did show up to compete but were completely ignorant of the Event content would perform poorly. Lastly, students cannot receive a negative score, which argues against a model where the y-intercept could go negative.

4.2.6.1 Regional Team Regression

As was shown in section 4.2.3, scoring within the grade book for an Event is *relative*; hence Regional Team Selection uses normalized scores for all Events to make sure that no Event is advantaged over another due to grade inflation in the grade book. However, performance results from the Regional Competition are *absolute* in the sense that the results within an Event are comparable across Events. Comparing *relative* scores to *absolute* scores doesn't work well, so for purposes of doing regressions, the Regional results are normalized using equation (21) and scaled to the range of scores from the grade book for each Event using only the scores of the students on teams in that Event from the time of the Regional Team Selection[127].

Normalizing the Regional scores to be in the same range as each of the Events' grade book scores allows the regression to be run such that all Events are handled equally. Take a case where there are two Events A and B. Assume that Event A's students are all scored high and Event B's students are all scored low in the grade book. At the Regional Competition, Event A's students perform poorly, and Event B's students perform well. It's difficult to run a useful regression on such mismatched data, where low grade book scores result in high performance and high grade book scores result in low performance, especially when other data from the gradebook may have results that are more in-line with common sense expectations. Normalizing the relative scores for each Event in the grade book and normalizing the Regional results to that same range establishes a base from which a regression can be run with sensible results.

In Table 16:, there is correlation between the relative placements of students within an Event in the grade book and the relative performance of students within an Event at Regional. Running the following regression twice provides the relative strength of student contributions in Table 25::

$$Z_i = \beta_0 + \beta_1 X_i + \beta_2 Y_i \qquad (25) \quad .$$

Where,

[126] Usually a student will take on a new Event after a round of Team Selection in order to fully staff an Event on their team that did not have a full complement of students after Team Selection was complete. This is often done by students to support a student who otherwise would have had to compete solo in that Event.

[127] That is, the composite scores that were used for the Regional Team Selection.

Z_i – is the performance of an Event pair at the Regional competition

X_i – is the grade book score of the higher-graded student in an Event pair

Y_i – is the grade book score of the lower-graded student in an Event pair

β_0 – is the intercept (forced to zero for this regression)

β_1 – is the X Coefficient or weighting for the higher-graded student

β_2 – is the Y Coefficient or weighting for the lower-graded student

Table 25: shows the two regressions that have been run on the available Regional data. The data was used in aggregate as well as in Event category data subsets. Below, the gray rows are discussed first, then the green rows.

Table 25: Regression Results: Regional

Data Set	Zero	X Coeff	Y Coeff	Inter cept	Scaled X	Scaled Y	Data Points	% 0's
All	N	0.636	0.288	0	0.688	0.312	636	
Build	N	0.640	0.211	0	0.752	0.248	124	
Study	N	0.674	0.277	0	0.709	0.291	298	
Lab	N	0.599	0.327	0	0.646	0.354	214	
All	Y	0.820	0.072	0	0.919	0.081	719	11.5%
Build	Y	0.876	-0.070	0	1.086	-0.086	149	16.8%
Study	Y	0.785	0.147	0	0.842	0.158	333	10.5%
Lab	Y	0.818	0.061	0	0.931	0.069	237	9.7%

The gray tinted rows at the bottom of the table used the full data set that includes all the zero secondary student score entries. The gray rows show that 9% to 17% of the data points in each category have a zero as the secondary student's score, which tends to increase the reliance of the regression on the primary student's score, thus increasing the X-coefficient and decreasing the Y-coefficient. For the Build Events, the Y-coefficient is actually negative, suggesting that the students competing in Build Events would be better off competing as solo students. This aspect of the model does not reflect the real world, since many of the devices require two students to set it up and/or operate the device, where one student alone cannot complete the task[128]. Similarly, for the Lab Events, the regression indicates that the high student accounts for over 90% of the pair's performance. This is also not reflective of the real world, since in a typical lab Event, one student focuses on the lab activities while the other focuses on the written (non-lab) portion of the test, so it is highly unlikely that the pairing's performance would be so dominated by one partner.

The green tinted rows in Table 25: show the regression on the same data but with the zero secondary student score entries removed. Here the X and Y coefficients are scaled into the weights that would be used on the data for the Regional Team Selection. The scaling is done such that:

[128] Recall that secondary student scores of 0 indicate that a student competed solo *OR* that a student had a new, inexperienced partner added to their pairing. Also note that *some* devices can be operated solo (but usually not as well as a pair can operate it). Of the Build Events, 1/3 of the zeros added a new partner and 2/3 operated solo.

$$X + Y = 1 \text{ and}$$
$$\frac{X}{Y} = \frac{Xcoeff}{Ycoeff} \qquad (26)$$

These regression parameters model the real world more closely, in that the secondary student still makes a significant contribution to the success of the pair, but the primary (higher-graded) student is the primary contributor to performance. Table 26: compares these parameters to the even weights used for Regional Team Selection and the skewed weights used for State Team Selection.

Table 26: Regression Weight Comparison

Data Set	Regression Scaled X	Regression Scaled Y	Regional X	Regional Y	State X	State Y
Build	0.75	0.25	0.5	0.5	0.8	0.2
Study	0.71	0.29	0.5	0.5	0.6	0.4
Lab	0.65	0.35	0.5	0.5	0.5	0.5

The Regression Scaled X and Y weights more closely match the State weights than the Regional weights, where the Build Events have heaviest weight on the primary student, Study Events the second heaviest weight on the primary student, and Lab Events have the least differential between the student weights. One possible contributor to the difference between the Scaled X and Y weights and the Regional X and Y weights is that the pairings of students on the Regional Teams are not all closely matched in ability, even though this was a stated goal of the Regional weightings. In Figure 31: and Figure 32:, it can be seen that for all Regional Teams approximately 43% of all Regional pairings are within 2 ordinal ranks of each other and 33% are 5 or more ordinal ranks apart. As with the secondary-zero data points, these more mismatched pairings tend to depend more on the primary student for the pair's performance. These mismatched pairs have not been removed from the regression data because: there are so many of them; they will continue to be present in future teams[129]; and even the closely-matched pairs have performance inconsistency issues[130] which cannot be objectively weeded out.

While not a perfect match with the rationale discussed in section 4.2.5, the regressions support the basic concept of the differing relative importance between the students in a pair in the different Event types and in the order of importance that was initially suspected.

[129] The mismatched pairs will also continue to be present in Team 1, which determines whether the school advances to the next level of competition or not.

[130] For example, a pair that has performed well in class and in Invitational competitions can unexpectedly bomb at Regional for many different reasons, where Build Events tend to have the highest degree of unpredictability due to breakage or other failure of their device. It is at best difficult to decide if poor performance was related to "bad luck" or was a result of actual poor performance. Similarly, a pair that has previously scored lower in Invitational competitions may do unexpectedly well at Regional, whether by "good luck" or a change in habits or preparation or other reason. Thus, it is too much of a subjective judgement call to filter points based on deviation of performance from expectations.

4.2.6.1.1 Sensitivity of Regression Weights

How much difference do different weights make in the actual composition of team members on a team?[131] Not surprisingly, for any one year, this depends a lot on the size of the student pool, the students' particular set of assigned Events, and the scores of the students in their Events. The larger the student pool, the greater the potential impact on the team members with a smaller change in weights.[132,133] Also, the closer student scores are within each Event, the more likely team composition would be affected by a change in weights. Since the pool size, Event sets, and student scores change every year, the variability of an effect on team composition with a change in weights also changes every year.

Taking the year 2020 as a representative year of data for Team Selection, the weights for the Team Selection algorithm can be modified and the resultant team members compared for Team 1. The 2020 Team Selection data has a pool of 81 students, with no two students having the same set of Events[134], which is typical of a data set during any competition year.

In Table 27:, the overall team rosters are compared for several different weighting schemes. The weighting schemes that are displayed are those that are commonly used or experimented with Team Selection, plus two more extreme weightings. Specifically, they are:

- Regional – the 50/50 equal weighting for all Events
- State – the unequal weighting for all Events originally postulated for selecting the most competitive team
- Regression – the weightings yielded from the regressions on actual competition performance data
- 80/20 – the most common suggestion for extreme weights for State Team selection
- 90/10 – a more extreme weighting scheme
- 99/1 – the most extreme weighting scheme possible

The green cells along the top of the table indicate the weighting schemes that yielded the exact same team. The green cells in the student roster rows indicate the students who appeared in all of the teams. The yellow cells indicate the students who appeared in most of the teams. The grey cells indicate the students who only appeared in the 'extreme' teams. And the grey/orange cells indicate the students who appeared in only one team.

---

[131] For purposes of variance, we are concerned only with Team 1. Once there is a difference in Team 1, there is a significant and progressively larger cascade effect on all teams formed after Team 1.

[132] Taking this to the extreme, with a pool of 15 students, the same team is formed regardless of the weights. As the pool increases in size, the potential for a change in team membership becomes greater due in large part to the greater number of potential team combinations that can be made from a larger pool. Likewise, as the pool becomes larger, it requires smaller changes to the weights to get a different team composition. For example, going to the other extreme of having an infinite pool of students and scores, a very small change in the weights would yield a different team composition.

[133] This also assumes that students are not set up with flights of Events, where the students are perfect replacements for each other. If students had flights, then the strictly dominant student in a flight would always be selected for the team regardless of weights. If there were not a dominant student in a flight, then weights would still impact team formation, however, the flights would effectively reduce the pool, requiring larger changes to the weight values in order to effect a change in team composition.

[134] There has only been one occurrence where two students had the same Event set. Those two students collaborated to have the same Event preference order and the same Availability. Given the RSD nature of the Event Assignment, it means that these students were processed in potentially very different order each round, yet still ended up with the same Event set. I think this speaks well of the efficacy and fairness of the system as it is applied to SO.

Table 27: Team Selection Weight Variance and Team Rosters

	Weight Scheme	Regional	State	Regression	80/20	90/10	99/1	Mod Regional
	Algorithm Score	2046.35	2081.52	2136.06	2188.2	2245.86	2320.66	2061.71
	Raw Score	4092.70	4092.70	4092.70	4092.70	3927.00	3770.60	4088.20
Weights	Study	50/50	60/40	71/29	80/20	90/10	99/1	60/40
	Lab	50/50	50/50	65/35	80/20	90/10	99/1	50/50
	Build	50/50	80/20	75/25	80/20	90/10	99/1	50/50
	ExpD	50/50/50	80/40/30	80/35/35	80/35/35	90/30/30	100/40/10	60/60/30
	WIDI	50/50	50/50	65/35	60/40	60/40	50/50	50/50
	Core Students	15	15	15	15	12	11	12
	New Students					3	4	3
	Student Roster (IDs)					2116	2116	
							2118	
		2120	2120	2120	2120			2120
		2147	2147	2147	2147	2147	2147	2147
		2151	2151	2151	2151	2151	2151	2151
		2274	2274	2274	2274	2274	2274	2274
		2297	2297	2297	2297	2297	2297	2297
		2301	2301	2301	2301	2301	2301	2301
		2311	2311	2311	2311			
								2317
		2326	2326	2326	2326	2326	2326	2326
						2332	2332	2332
						2336	2336	
		2389	2389	2389	2389	2389	2389	2389
		2393	2393	2393	2393			2393
		2411	2411	2411	2411	2411		2411
		2466	2466	2466	2466	2466	2466	2466
		2484	2484	2484	2484	2484	2484	2484
		2505	2505	2505	2505	2505	2505	
		2522	2522	2522	2522	2522	2522	
								2564

The Regional, State, Regression, and 80/20 schemes all yielded the exact same team over a wide range of weights from 50/50 to 80/20. The 2020 data is atypical in that the Regional and State weightings typically do not yield the same team, but rather usually differ by one to three students. The 90/10 weights usually differ from the State team by one to three students, as is the case with the 2020 data. The 99/1 weights will frequently differ by zero or one student from the 90/10 weights.

Another way to compare the teams made from the different weightings is to look at the sum total raw scores for each team. That is, sum the student scores for the Events in which they are competing without any weighting. This result is shown in the "Raw Score" row of Table 27:. Note that the Regional weights of 50/50 for all Events means that this result also *maximizes* the raw score, thus no other weighting can exceed the raw score achieved by the Regional team. The more extreme weightings will (almost) always yield lower raw scores than the Regional weights because they focus on the primary student's score for each Event, and the secondary student's score contributes progressively less to the objective function as the weights become more skewed. The result tends to favor the highest scoring student in each Event, where the secondary student choice in each Event is more driven by whether or not that secondary student scores highest in one or more other Events. Thus, the more skewed the weights, the more likely there will be a large disparity between student scores (ability) in each Event. These raw scores show that the more skewed the weights, the lower the raw score for the team. For example, the difference in raw score between the Regional team and the 99/1 team is 322 points (7.9%), which is about equivalent to the difference in team score between Team 1 and Team 2 in a tiered set of teams with a 90+ student pool. This would tend to indicate that highly skewed weightings (certainly beyond 80/20) are probably not wise in most competitive situations.

Interestingly, the Mod Regional team is different from the Regional team by 3 students even though the main difference in the weight schemes is for the Study Events, where the weight was changed by a relatively modest amount from 50/50 to 60/40.[135] Looking in detail at the students involved and their Events, there were 10 Events affected by the change in student roster. Of these 10 Events, on the Regional team 8 Events were affected only in their secondary student, and two Events were affected with their primary student; and on the Mod Regional team, 9 Events were affected only in their secondary student, and one Event was affected with the primary student. The net change in *unweighted* team score between the two teams was a 4.5 point decrease for the Mod Regional team compared to the Regional team. When looking at the pairs in these 10 Events, the net effect was to make some Events stronger, some Events weaker, and other Events stayed about the same. Specifically, with the Mod Regional team, one build Event and one Study Event became significantly stronger while one Lab Event and two Study Events became much weaker. This is similar to the circumstance of running more iterations of the Event Assignment, where the result is merely shifting around which students are better off with no significant change to the overall pool's surplus. Given the fact that Build Events don't need two strong students to perform well, it's easily arguable that the Mod Regional team is a weaker team than the Regional team since the strength of the second strong student in the Build Event is wasted. This reinforces the importance of an unequal weighting of Build Events for higher levels of competition, as those unequal weights will inhibit slotting two strong Build Event students together at the expense of Study or Lab Events.

Overall, the results with the 2020 data mimic that of prior years, where differences of 20% in the weight values typically result in a roster change of between 1 and 4 students, and 30%+ weight changes result in roster changes of between 3 and 6 students. Small changes of 5% from the 'standard' weights usually resulted in no change to the team roster. In these prior years, the Head Coaches were allowed to request changes in Team Selection weights "just to see what the team would look like" for the State Teams.

Based on this, the weights can be changed by a significant amount from the values calculated by the Regression and still yield the same or almost the same team. This indicates that a pool of students between 60 and 150 yields relatively stable results over a fairly wide range of weights. Thus allowing the Head Coaches a range-limited ability to set the Team Selection weights would provide a perceived level of control without substantially altering the outcome pointed to by the regression analysis.

4.2.6.1.2 Data Points and the Prediction Plane

An interesting way to look at the regression data is to plot the student-pair data points against their actual performance above and below the plane described by the regression coefficients.[136]

[135] There was no change to the ExpD students, so the ExpD weight differences had no impact on the roster.

[136] A rotating 3D plot of the points on a prediction plane is more fun to view, but this can't be done in a document…

Above the plane indicates that the pair outperformed versus the pair's expected performance, and below the plane indicates the pair underperformed versus the pair's expected performance. The following figures show the data points that lie above or below the plane for the four categories on which the regressions were run: All Points, Build, Study, and Lab. In addition, the table following each plot-pair indicates how many points lie above the plane and below the plane. The variegated color of the plane reflects the prediction value for the X-Y pair at each location, hence the plane is colored from dark blue (minimum performance) near the origin to yellow (maximum performance) in the top right corner. Each point is colored in its actual performance on the same scale from dark blue to yellow. The plot on the left shows the points which outperformed expectations, and the plot on the right shows the points which underperformed expectations. Recall that the Regional data points are normalized and scaled to show relative performance since the nature of the grade book prevents being able to compare absolute performance across Events.

Figure 15: Regional Regression Plane Plots: All[137]

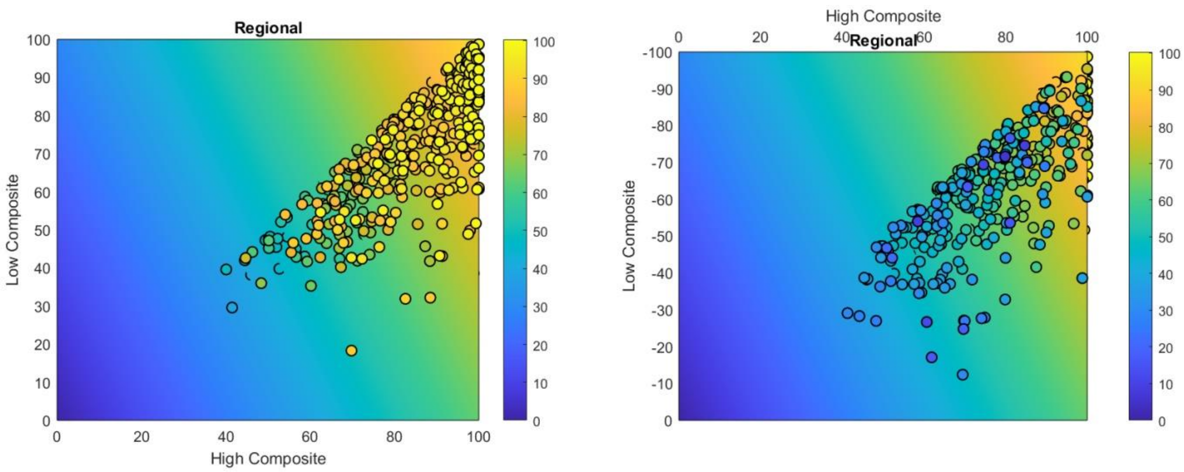


Table 28: Regional Regression Points Over and Under: All

Total Points	Points Above	Points Below
636	342	294

Figure 15: and Table 28: look at all the student-pair data points and appear to show that student-pairs at Regional should out-perform their predicted performance by a 1.16:1 margin over the pairs who under-perform. From the plot densities, the Above plot shows that the pairs that out-perform are clustered more in the upper-right corner and along the upper-right portion of the diagonal, where there is a high-scoring primary student (right side of plot) frequently matched with a student closer in score (the diagonal). The Below plot's density tends to be parallel to and a little below the diagonal and closer to the 60-80 band on the x-axis, where the primary student is slightly lower scoring and the secondary student has a greater differential in score from the primary than the pairs in the Above plot.

The general trend in the Above and Below plots in Figure 15: appear to conform with the common-sense reasoning that two well-matched higher-scoring students would *likely* perform higher than expected as well as perform higher in an absolute measure than two mismatched

[137] Note the Below plots have inverted 'low' axis in order to get the data to display 'aligned' with the Above plots.

lower-scoring students (who would likely perform worse than expected). This trend also appears to be buoyed by the trends in the category-specific plots following. However, the notion that teams that out-perform outnumber teams that under-perform by 1.16:1 does not hold up in the category-specific plots. This indicates that while running a regression on all the data points can provide some insight into a data trend, there are sufficient differences in characteristics of Events in each category to warrant analyzing the categories separately. This also supports the argument for using different weights for each Event category for team selection.

Figure 16:, Figure 17:, and Figure 18: and their accompanying table tallies show that the density of points for the Above plots tend to be toward the upper-right corner of the plots and closer to the diagonal. The Below plots tend to have their density around and below 80 (on the x-axis) and a little further below the diagonal than the Above plots. This is in agreement with the plots for All points. The category specific plots differ from the All plots in that the distribution of points above and below the plane are nearly 1:1 for Build and Lab Events, whereas Study Events show a ratio of above to below points of 1.2:1 instead of the 1.16:1 seen with All Events.

The general trends here are that pairs are:

- about equally likely to outperform as underperform
- the more closely pairs are matched, the more likely they are to outperform
- the more mismatched pairs are, the more likely they are to underperform
- higher scoring pairs are more likely to outperform

Figure 16: Regional Regression Plane Plots: Build

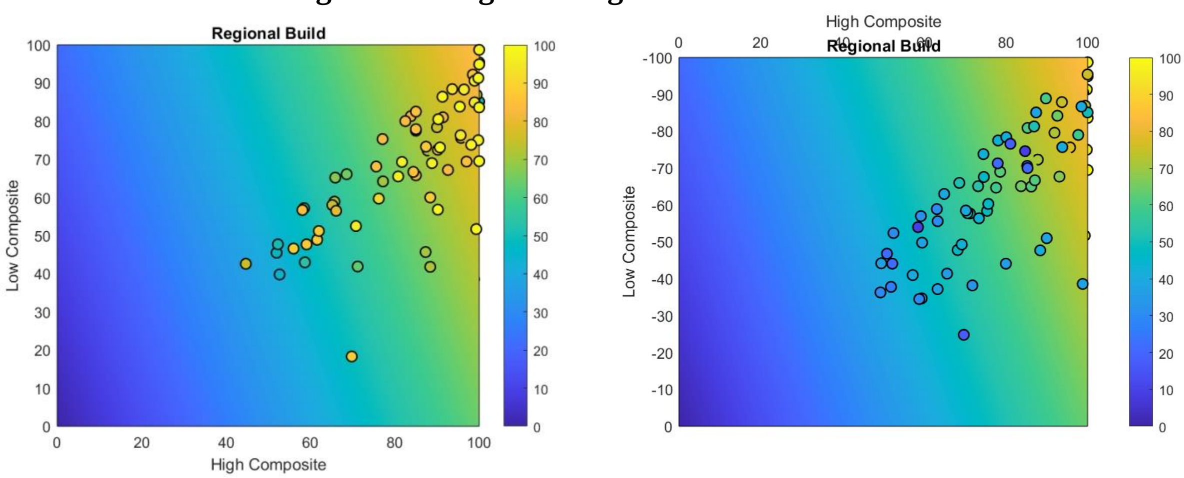


Table 29: Regional Regression Points Over and Under: Build

Total Points	Points Above	Points Below
124	62	62

Figure 17: Regional Regression Plane Plots: Study

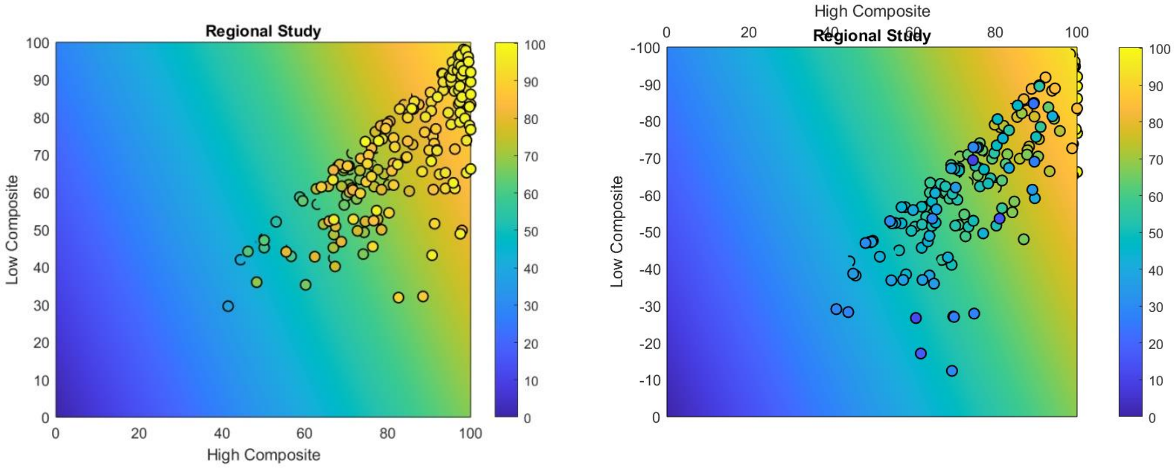


Table 30: Regional Regression Points Over and Under: Study

Total Points	Points Above	Points Below
298	163	135

Figure 18: Regional Regression Plane Plots: Lab

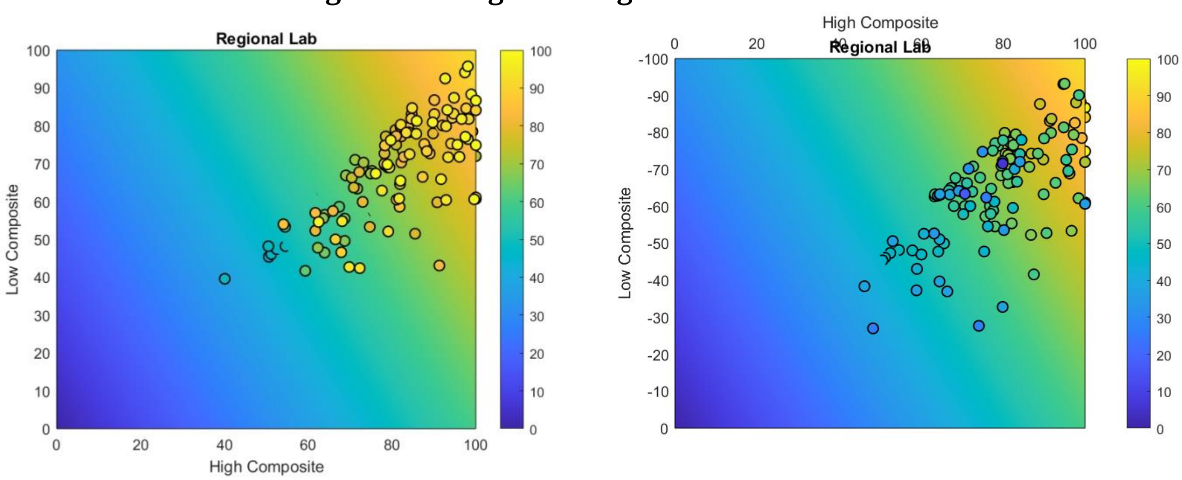


Table 31: Regional Regression Points Over and Under: Lab

Total Points	Points Above	Points Below
214	106	108

4.2.6.1.3 Data Points and Deviation from the Prediction

Another interesting way to view the data is to plot the deviation of the performance of the student-pair from the prediction from least deviation to greatest deviation, both above and below prediction. It is also interesting to see which pairings tend to deviate further from the prediction. The following figures plot the data points and their deviation from the prediction, with the blue line showing those student-pairs who performed better than the prediction and the red line showing those pairs who performed worse than the prediction. The table following each plot takes the data points and sorts them from high to low by the primary student's score and shows the aggregated counts of the number of pairs who performed better or worse than predicted broken down into four groups, where the first group contains the highest-scored primary students and the last group contains the lowest-score primary students.

From section 4.2.6.1.1 it's clear that the All deviation analysis is not the best way to look at the data, but it can still provide some conceptual insights. At the left end of Figure 19:, for about one third of the data points, the positive and negative deviations are about the same in magnitude and number. Nearer to the right end of the graph, the negative deviations become significantly larger than the positive deviations, which would indicate that it's easier for a pair to bomb in their performance than to greatly outperform expectations.[138]

From Table 32:, pairs with primary students ranked in the top 2 quartiles are more likely to outperform than underperform, with the greatest degree of outperformance experienced by those pairs in the top quartile. Pairs in the 2nd quartile are likely to experience the largest drop in performance. How do these apparent trends hold up when the data is sorted by Event categories?

Figure 19: Regional Regression Deviation: All

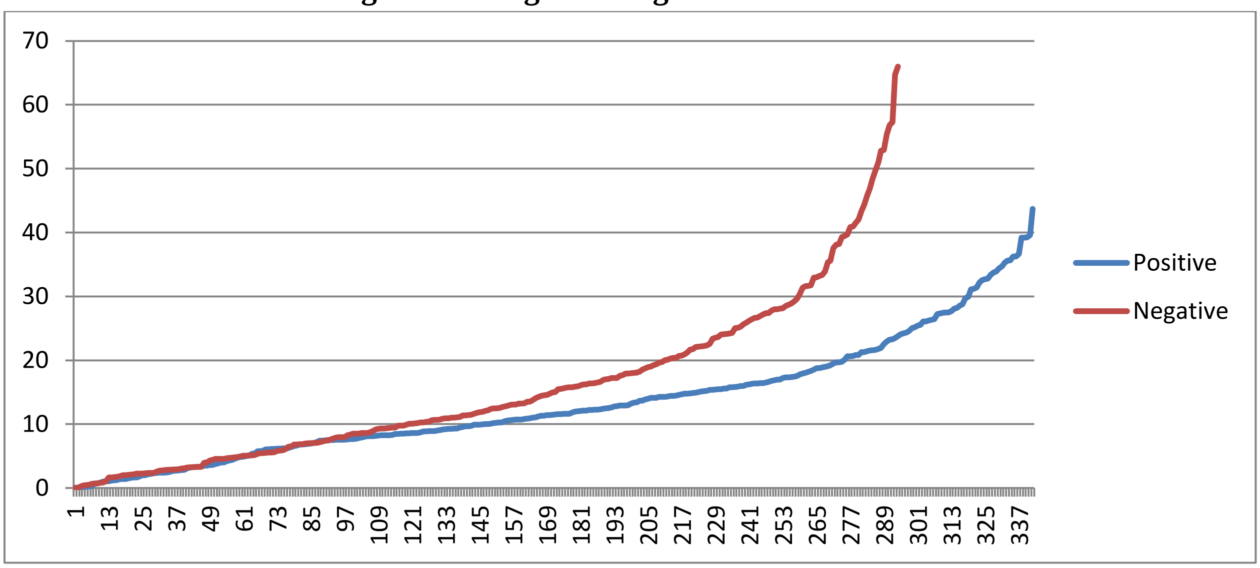


Table 32: Regional Regression Deviation: All

Quartile	Count	Pos	Neg	Average	Avg Pos	Avg Neg	Med	Med Pos	Med Neg
All	636	402	234	4.6	15.8	-14.5	6.7	15.3	-11.1
1	159	121	38	8.5	15.7	-14.6	15.0	16.1	-11.7
2	159	101	58	1.9	14.3	-19.6	6.2	14.0	-15.4
3	159	90	69	3.6	17.4	-14.5	3.0	15.8	-12.0
4	159	90	69	4.6	16.1	-10.3	2.1	13.2	-9.4

[138] Note that this statement does not contradict the statements in the preceding and following paragraph which state that higher scoring pairs have a higher likelihood of out-performing vs. lower scoring pairs. The statement about bombing describes the *degree* by which a team underperforms, not the likelihood of underperforming. This means that if a pair does underperform, there is a higher likelihood that the pair will underperform significantly versus underperforming by a little; and that a pair that outperforms has a low likelihood of outperforming by a large margin and a higher likelihood of outperforming by a little.

Figure 20: and Figure 22: cover the Build and Lab Event categories and are similar to each other. Both show that the magnitude and number of positive and negative deviations track closely for almost all of the graphs. Only at the right-end where the largest deviations are recorded do the greatly underperforming pairs exceed the highly outperforming pairs in degree of deviation. These large deviations in performance in the Build and Lab Events are most likely those incidents where the device failed (or the device ran perfectly for the first time), or the pair contaminated their chemical samples (or all the lab tests went right the first time of testing them). In Table 33: and Table 35:, it can be seen that this data tracks the All table, where student pairs in the top 2 quartiles are more likely to outperform than underperform. However, while Build Events can expect the greatest underperformance drop in the 2nd quartile, the Lab Events can expect the largest performance drop in the 3rd quartile.

For the Study Events in Figure 21: and Table 34:, the data tracks a little more closely to the All graph. The positive and negative plots start diverging about ¼ of the way along the graph and the negative deviations significantly exceed the positive deviations in degree. However, it is only in the 1st quartile that pairs are more likely to outperform than underperform while it is in the 4th quartile that pairs tend to achieve their greatest amount of over-performance. In the 2nd quartile, about an equal number of pairs outperform as underperform, and the 2nd quartile is also where pairs are likely to have their largest degree of underperformance.

Figure 20: Regional Regression Deviation: Build

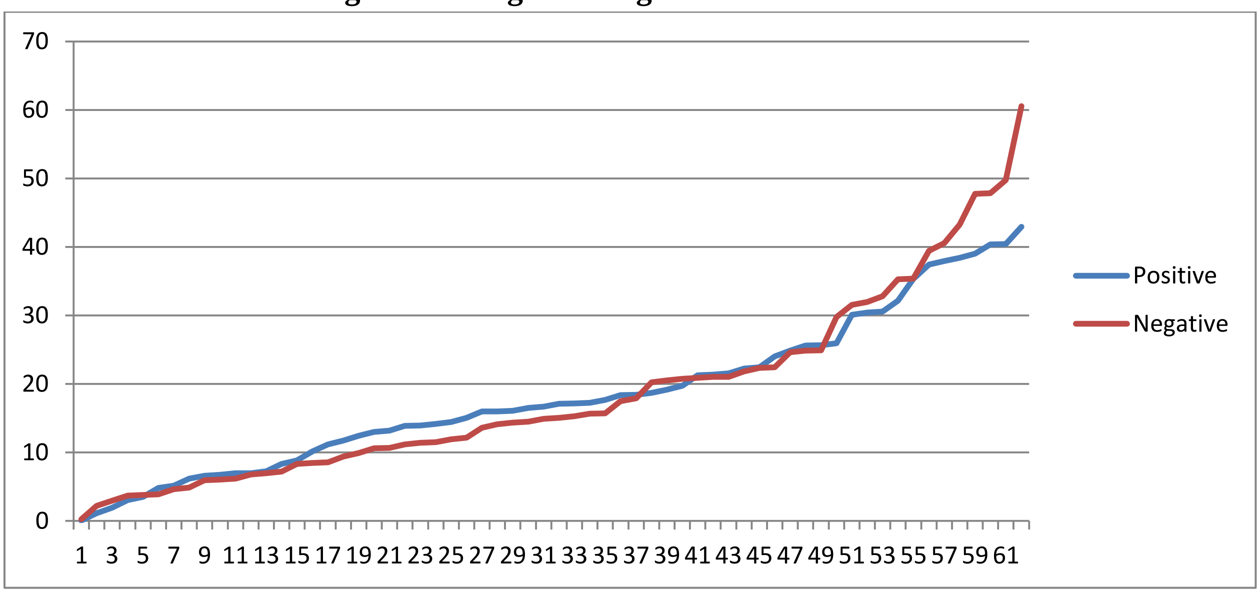


Table 33: Regional Regression Deviation: Build

Quartile	Count	Pos	Neg	Average	Avg Pos	Avg Neg	Med	Med Pos	Med Neg
All	124	62	62	-0.1	18.3	-18.5	-0.1	16.9	-15.0
1	31	20	11	3.1	15.1	-18.6	10.2	16.3	-11.9
2	31	18	13	-0.5	16.5	-23.9	6.2	15.1	-21.0
3	31	10	21	-4.2	23.3	-17.3	-7.2	24.1	-14.3
4	31	14	17	1.1	21.7	-15.8	-4.8	19.3	-14.9

Figure 21: Regional Regression Deviation: Study

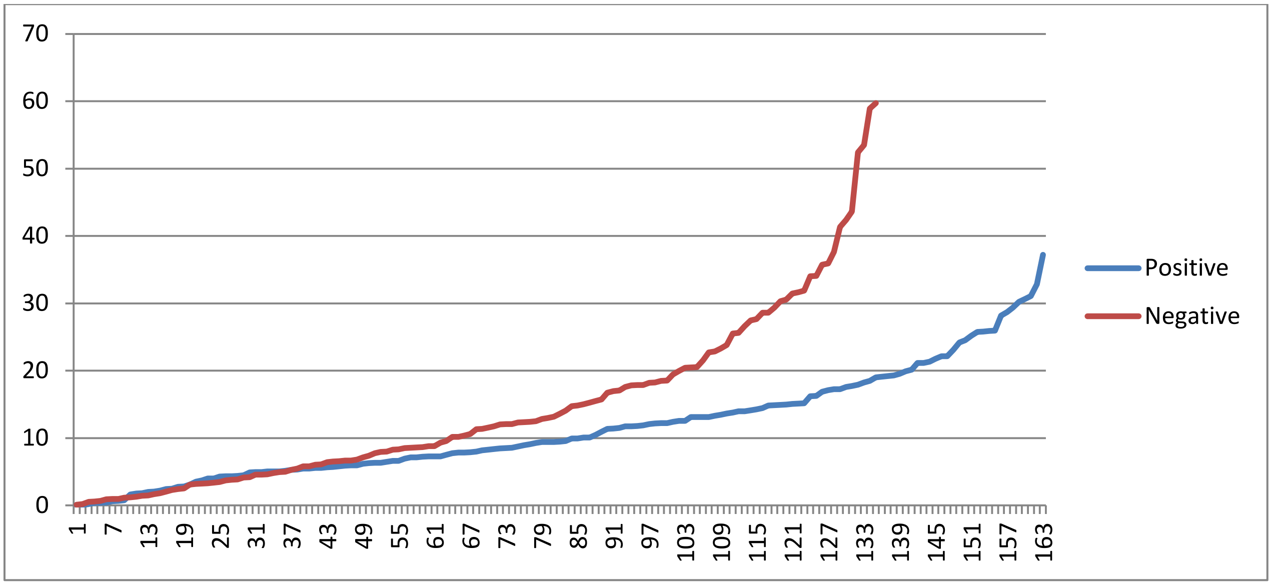


Table 34: Regional Regression Deviation: Study

Quartile	#	Pos	Neg	Average	Avg Pos	Avg Neg	Med	Med Pos	Med Neg
All	298	163	135	-0.3	11.4	-14.4	2.1	9.5	-11.3
1	75	61	14	4.2	8.4	-14.5	6.5	7.8	-10.6
2	75	36	39	-3.8	11.3	-17.7	-0.9	11.8	-12.3
3	75	40	35	-0.8	12.9	-16.6	0.4	12.8	-14.7
4	73	26	47	-0.6	16.2	-9.9	-3.5	15.0	-8.6

Figure 22: Regional Regression Deviation: Lab

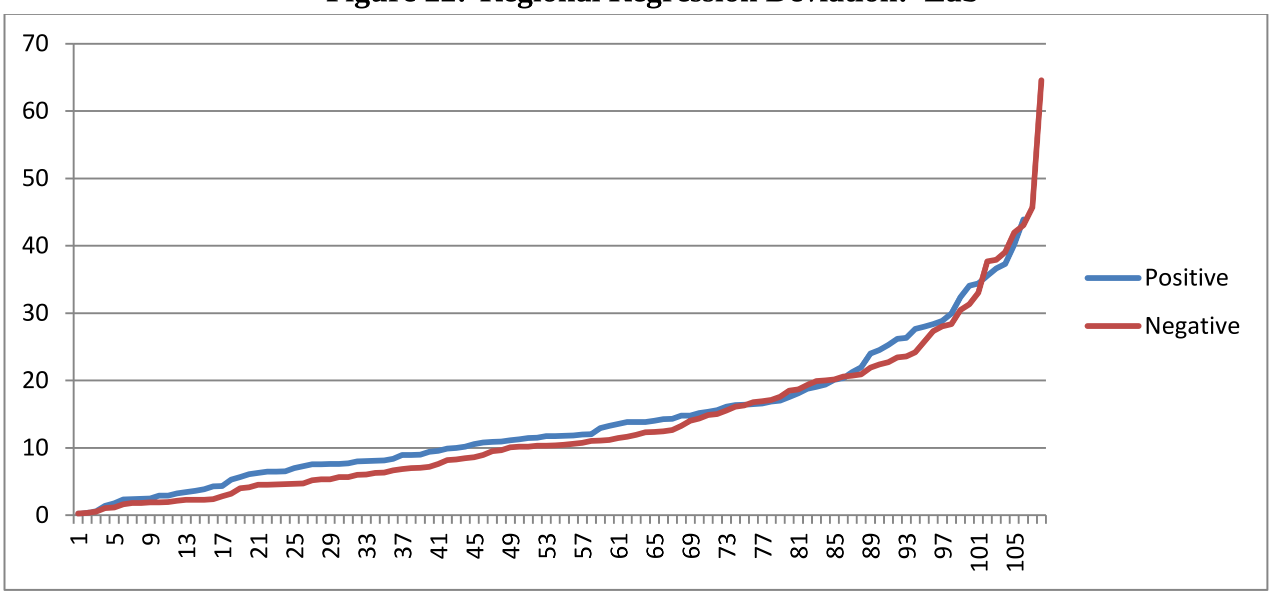


Table 35: Regional Regression Deviation: Lab

Quartile	Count	Pos	Neg	Average	Avg Pos	Avg Neg	Med	Med Pos	Med Neg
All	214	106	108	0.1	13.9	-13.5	-0.3	11.8	-10.4
1	54	31	23	-0.6	10.5	-15.6	3.2	11.1	-15.0
2	54	29	25	2.0	12.8	-10.6	4.9	10.6	-7.0
3	54	23	31	-3.6	15.8	-18.0	-4.6	14.8	-12.3
4	52	23	29	2.6	17.8	-9.4	-1.5	11.8	-8.9

4.2.6.2 State Team Regression

As was shown in section 4.2.3, scoring within the grade book for an Event is relative, while scoring from Invitational, Regional, and State competitions are absolute measures. The composite scores for students that are used for State Team Selection are composed of 60% to 90% of Invitational and Regional scores, where the grade book score from the Event coach is normalized across all Events so that no one Event has an advantage over another Event due to grade inflation. As these State composite scores[139] are predominantly calculated from the competition scores, they should be better predictors of State performance than the subjective grade book is for Regional performance. For purposes of the analysis of State results, the students' State composite scores and the State results are both considered absolute scores and can be compared without needing to do any scaled normalization of the data.

As with the Regional regression process, the State data is reviewed in whole and in Event category sub-groups of Build, Study, and Lab. There was only one point of zero-data in the State data (Build category), and that point was removed for the regression analysis.

Table 36: Regression Results: State

Data Set	Zero	X Coeff	Y Coeff	Inter cept	Scaled X	Scaled Y	Data Points
All	N	0.599	0.280	0	0.681	0.319	141
Build	N	0.738	0.084	0	0.897	0.103	36
Study	N	0.910	-0.011	0	1.012	-0.012	73
Lab	N	0.202	0.630	0	0.243	0.757	32

The regression coefficients for the State data are like a mix of the Regional regression results. The State All regression mimics the Regional All regression with the zero-data removed (top green row of Table 25:). The State Event category regressions look similar to the Regional Event category regressions with the zero-data included (gray rows of Table 25:). As with the discussion of the gray rows of the Regional regression in section 4.2.6.1, the Study and Lab regression coefficients from the State data don't reflect the real world, in that it is highly unlikely that the secondary student in a Study Event would detract from the pair's performance or that the secondary student in a Lab Event would contribute 3 times more to the performance of the pair than the primary student would.

Note that a number of different regression formulas were checked to see if there might be a formulation that yielded reasonable results for both the Regional and State data. None of the

[139] These are the scores used by the Team Selection algorithm for selecting the State team.

formulas tried yielded a better result for the Regional data than was obtained using the simple "X + Y" formula. Likewise, no improvement in the results for the State data was found either. The formulas that were tried included:

$Z \sim \beta_1 X$	(27) .
$Z \sim \beta_2 Y$	(28) .
$Z \sim \beta_1 X + \beta_2 Y$[140]	(29) .
$Z \sim \beta_1 (X+Y) + \beta_2 (X-Y)$	(30) .
$Z \sim \beta_1 X + \beta_2 (X+Y)$	(31) .
$Z \sim \beta_1 X + \beta_2 (X-Y)$	(32) .
$Z \sim \beta_1 X + \beta_3 XY + \beta_2 Y$	(33) .
Z**2~ β_1X**2 + β_2Y**2	(34) .
Z**2~ β_1X**2 + β_3XY + β_2Y**2	(35) .

The data was also cleared of potential outlier points to see if the regression results would improve, but to no avail. The outliers were filtered out by using the common practice of removing any point 1.5 times the interquartile range above the third quartile or below the first quartile. While a few points were removed, there was no significant impact on the regression results.

Overall, it appears that the State data has too few points to be as useful as the Regional data and that the State data is much more random than the Regional data. This can be seen in the next two sections looking at the data and the prediction planes.

4.2.6.2.1 State Data Points and the Prediction Plane

Looking at all the points in Figure 23: and Table 37:, the Above points outnumber the Below points by about 1.6:1, which is much higher than the Regional data. However, unlike the Regional data, there does not appear to be any significant difference in clustering between the Above and Below points; that is, both appear to be clustered more between 80 and 100 on the x-axis and both are close to the diagonal, and both appear to have a similar scattering of outliers away from the diagonal. There don't appear to be any visual patterns or trends in these plots.

For the Build Events in Figure 24: and Table 38:, the number of points is small at 36. The ratio of Above to Below is 2:1. There is a hint of a pattern similar to the Regional data in that the Above points have a denser grouping around 90 on the x-axis and close to the diagonal, and the Below points have a grouping around 80 and a little below the diagonal. However, the pattern is not as clear as with the Regional Build data.

For the Study Events in Figure 25: and Table 39:, the ratio of Above to Below is nearly 2:1. There are enough points to see some clustering; however, the clustering appears to be located

[140] It should be noted that equation (29) is equivalent to equation (25), and that equations (30), (31), and (32) are functionally equivalent to equation (29). Equation (30) actually has better statistical correlation than equation (29). This is likely because the data is represented in a different form, being the combined value of X+Y and the difference between them (X-Y). This makes 'sense' in that student performance is related to the combined strength of the two students as well as the difference in ability between the two students.

nearly identically for both: upper-right corner and close to the diagonal. Again, there are no significant patterns that differentiate the data.

Figure 23: State Regression Plane Plots: All

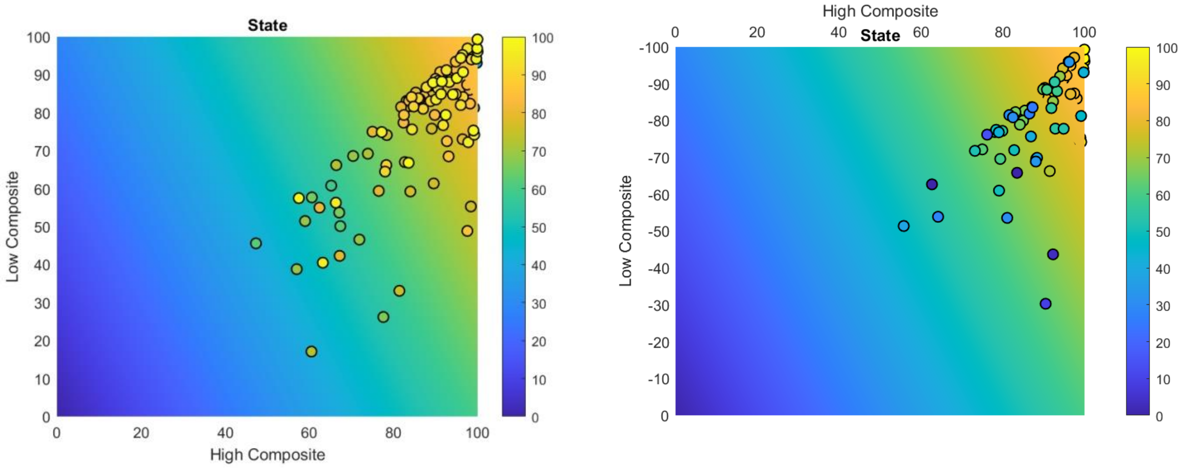


Table 37: State Regression Points Over and Under: All

Total Points	Points Above	Points Below
141	87	54

Figure 24: State Regression Plane Plots: Build

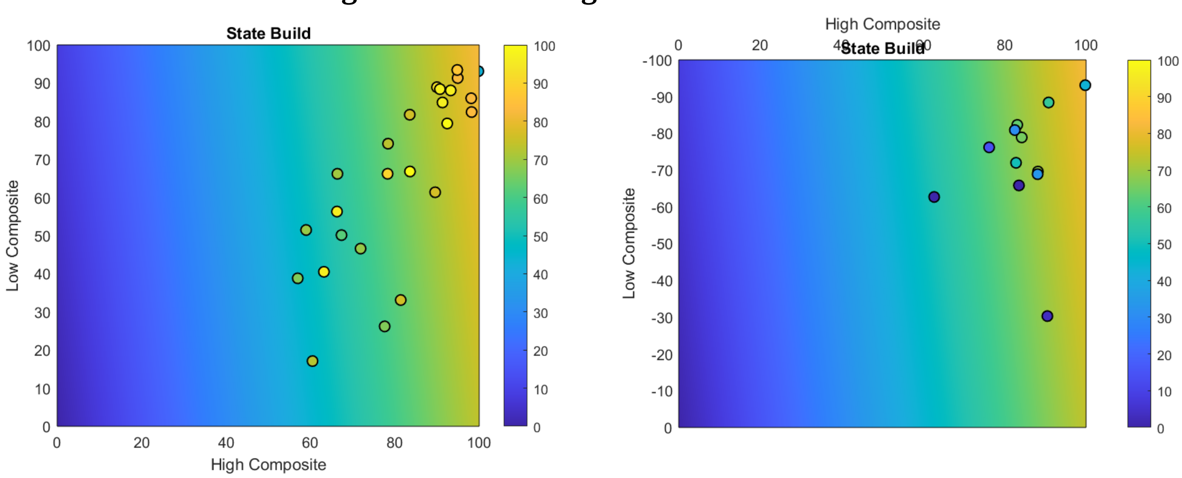


Table 38: State Regression Points Over and Under: Build

Total Points	Points Above	Points Below
36	24	12

Figure 25: State Regression Plane Plots: Study

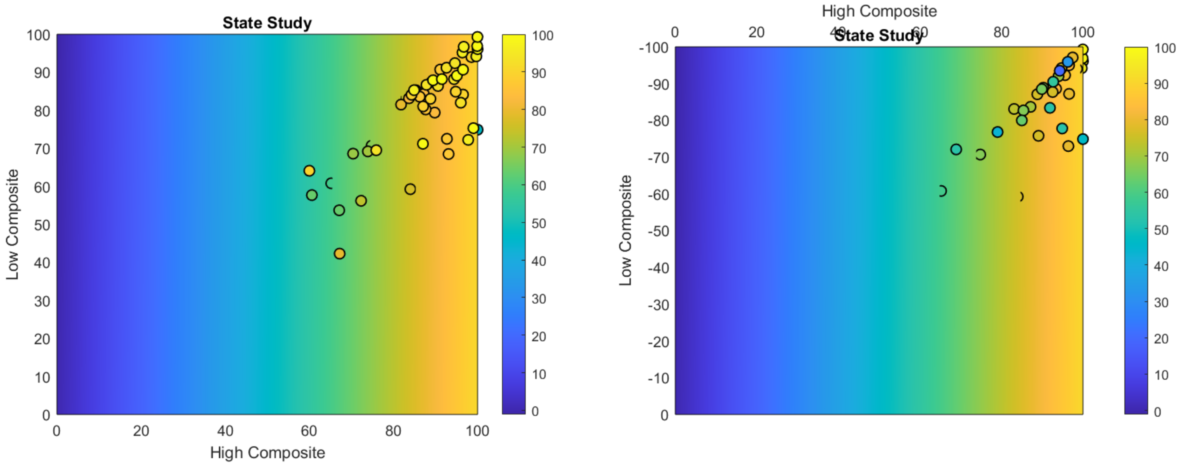


Table 39: State Regression Points Over and Under: Study

Total Points	Points Above	Points Below
73	46	27

Figure 26: State Regression Plane Plots: Lab

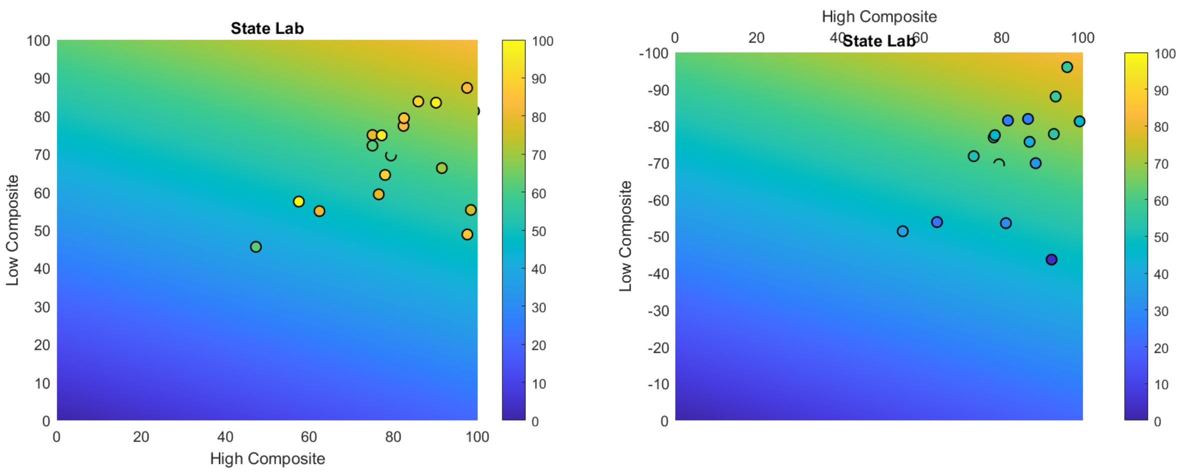


Table 40: State Regression Points Over and Under: Lab

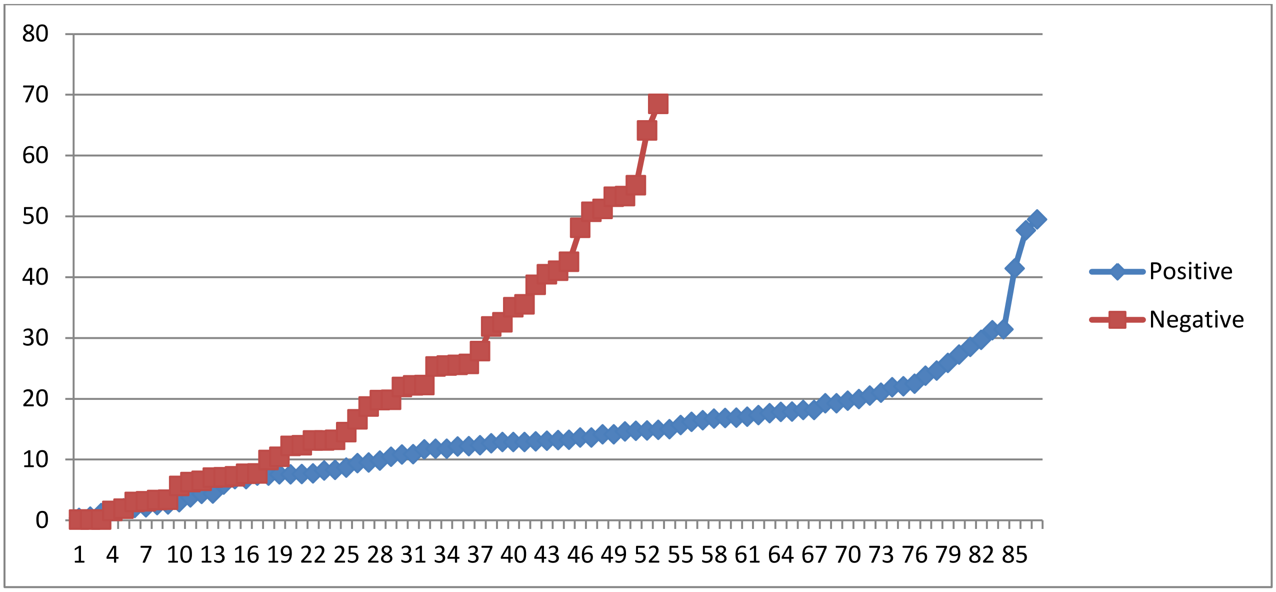
Total Points	Points Above	Points Below
32	17	15

For the State Lab data in Figure 26: and Table 40:, there are not enough points to conclusively show any pattern in the data. The distribution and clustering of the Above and Below points are very similar and without pattern in this plot.

4.2.6.2.2 State Data Points and Deviation from the Prediction

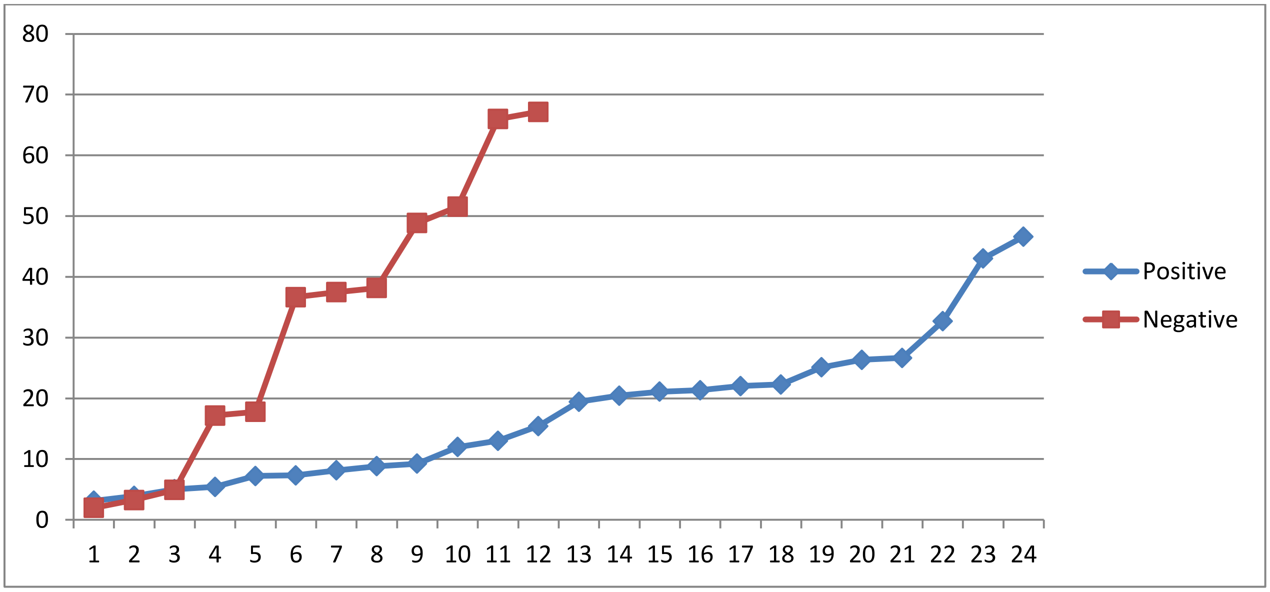
The following graphs show the degree of deviation of the State performance from the predicted performance. While the State Regression coefficients may not contribute to an accurate model of student performance predictions at State, the graphs do provide some insight into the data and the real world student performance at the State competitions.

Figure 27: State Regression Deviation: All



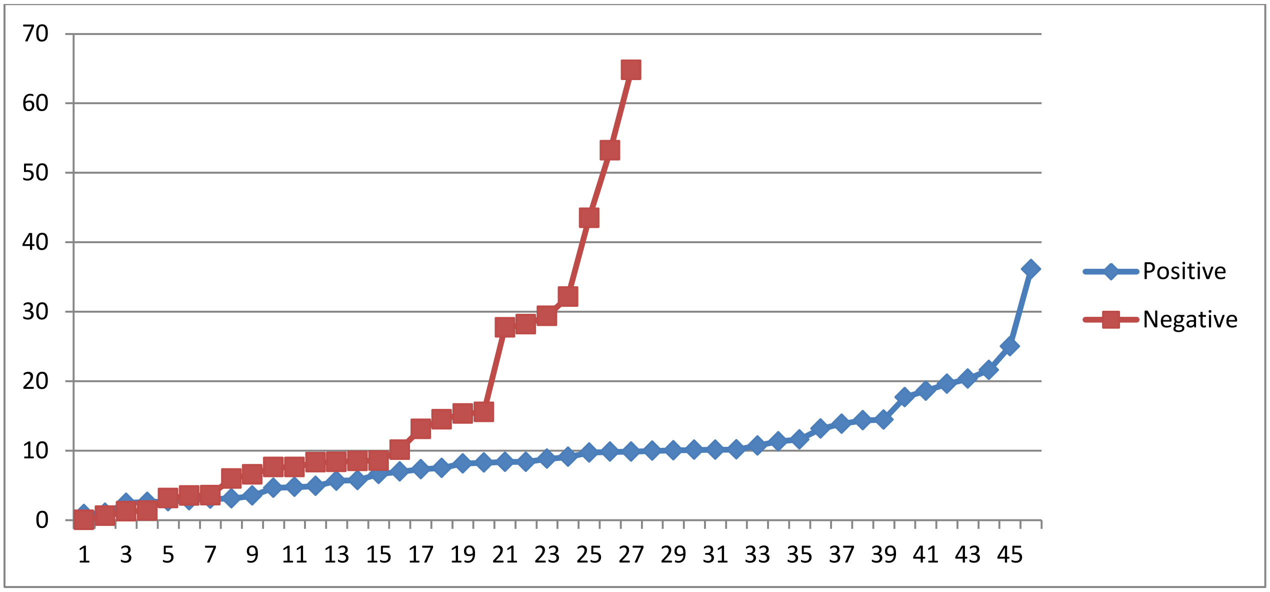
In Figure 27:, as with the Regional All data, it's easy to see that a team is more likely to bomb on a State competition test than it is to greatly outperform expectations. Also, the over- and underperform deviations track in degree for about 20% of the data points before the underperforming deviations diverge significantly from the outperform points.

Figure 28: State Regression Deviation: Build



In Figure 28:, the large negative deviation in the State Build data can be attributed in part to the last minute parental interference on student devices which causes the devices to fail or perform much worse than before the parent intervened. While the number of incidents of parental interference concerns around 20% of the build Events, the net negative effect is significant.

Figure 29: State Regression Deviation: Study

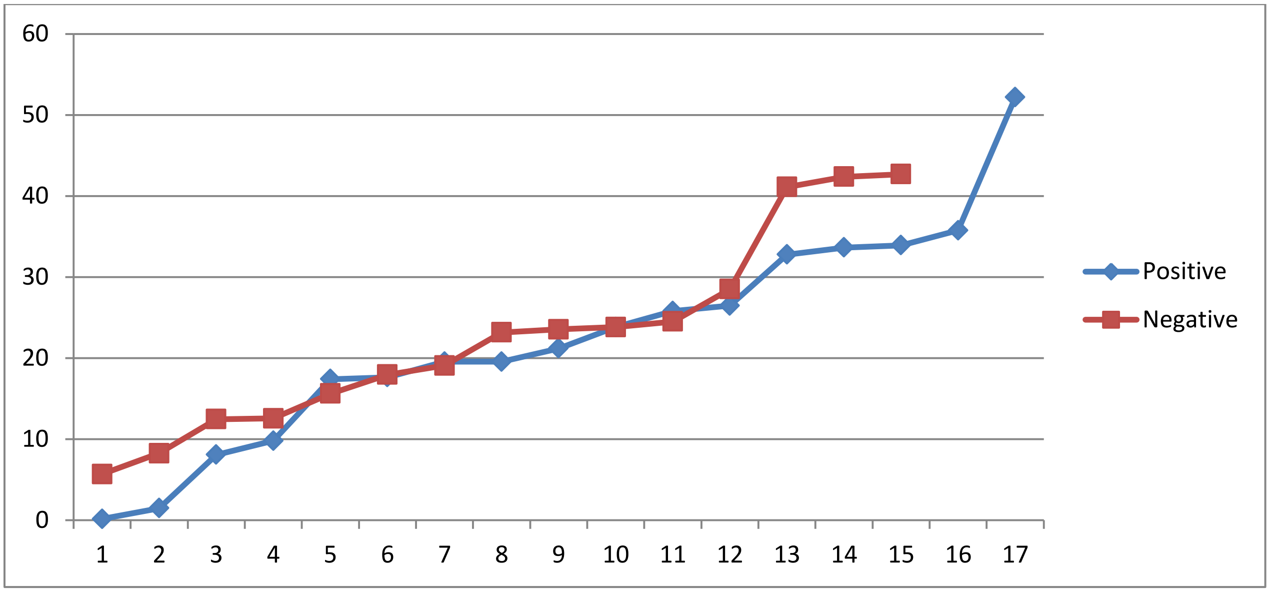


In Figure 29:, the State Study Event data is shown. Here, the jump in negative performance can be attributed in part to Event rule changes between Regional and State. For State, some Events list additional material and topics in the rules over what was covered in the Regional competitions. If students did not prepare for this additional material, it would tend to negatively impact their performance. Also, there have been a couple of Events each year at State where the students reported that some of the test questions covered areas that were not in the Event rules at all. There are a number of college students who volunteer to create the State tests and some of the college students appear to have a tendency not to follow the topics in the rules closely or include more in-depth material from their college courses[141].

Figure 30: displays the State Lab data. The Above and Below lines are nearly identical to the Regional Lab data in Figure 22:. The two lines track each other closely over all the data points. There is nothing new or different in this State Lab data relative to the Regional Lab data.

[141] An example of this is one year, a series of questions on earth science had students look at silt samples collected from river mouths and then identify the river from which the silt came. However, there was no mention of silt or river mouth estuaries anywhere in the rules for the Event. The closest topic to this was coastal ocean currents, which, again, did not list silt as part of the topic.

Figure 30: State Regression Deviation: Lab



4.2.6.3 Regression Summary

The Regional Regression results suggest that stronger teams could be made by using skewed weights in the Regional Team Selection algorithm. However, this does not take into account the impact that using skewed weights for the selection of Regional Teams would subsequently have on State Team Selection. The most critical impact is the potential for Free Riders' composite scores to be elevated by their higher performing partners' scores and thereby be selected for the State Team. This is because the skewed weights tend to place 'specialist' students[142] on the team due to the heavy weight given to the primary student and low weight given to the secondary student. This can result in many unequal pairings on the Regional teams, where both members of the pair share the same competition scores. Those shared competition scores elevate the secondary student's scores in their lower performing Events, and make that student more attractive to the State Team Selection algorithm even though the student may not have had any performance improvement in those Events.

Though the regression results suggest that skewed weights would improve Regional Team Selection, the exacerbation of the Free Rider problem in the State Team Selection overrides making any change in the Regional Team Selection weights.

The Regional Regression results reinforce the use of skewed weights for State Team Selection. For the State Team, it is desirable to bring in specialist students as the Free Rider problem is not a concern after the State competition. The caveat here is that the Regional plots show that more closely matched students tend to have a higher probability of outperforming their predicted performance than do mismatched students; and that a pair with a very high performing primary student has a higher probability of outperforming than does a pair with a moderately high primary student.

[142] A 'specialist' student is one who performs highly in one of their Events, but does not perform well in any of their other Events.

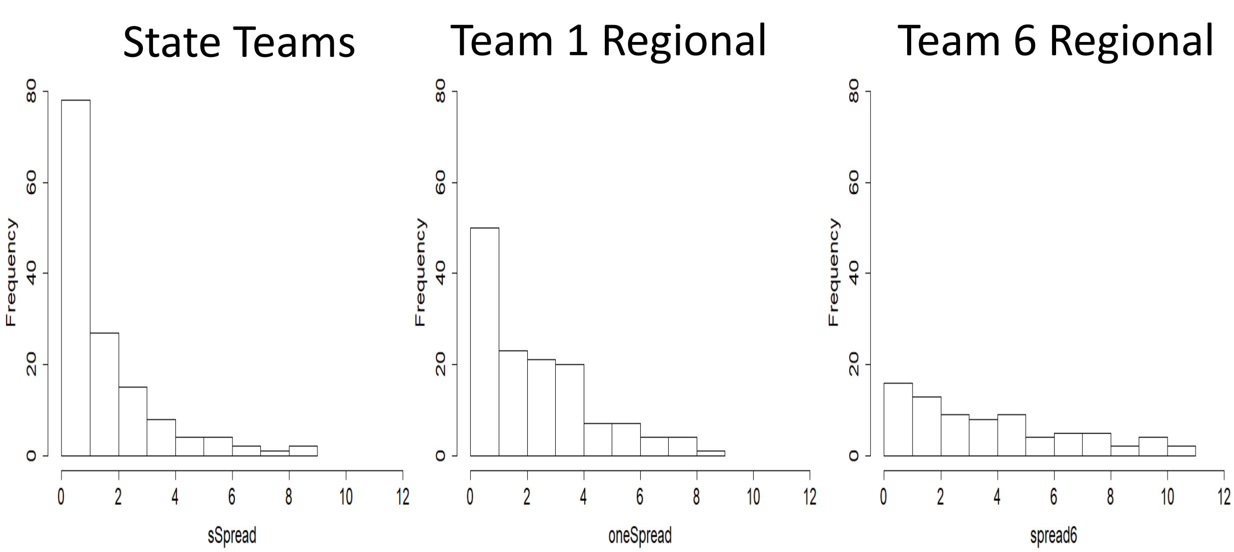
The State Regression results are inconclusive, as there appear to be too few points to arrive at meaningful results, in part because the State data points also appear to be more randomly associated with performance than the Regional data.

4.2.7 Partnership Pairing

One of the intended goals of Team Selection for the Regional Teams is to pair students of similar ability. Given the variability of grading by coaches, the granularity of those grades, and the very limited amount of 3rd-party assessments of student performance when Regional Team selection occurs, it is difficult to compare student ability based on their total grade book score. This is because the gaps between student scores, for the reasons previously discussed in section 4.2.3, are highly variable from Event to Event. For example, students in one Event may be distributed over a wide range of scores, while another Event's students might all be tightly clustered inside a very narrow range of scores. However, if we look at the students by their ordinal ranking in each Event, we know that the students in every Event who are closest in ability will be only 1 ordinal rank apart, regardless of their percentage score differences. Given the granular nature of team selection, where the algorithm must select one student or another student, it makes sense to use ordinal ranks rather than percentage score differences as a reasonable proxy to measure similar ability.

The histograms in Figure 31: were generated by plotting the data on team selection for all the State teams, and Regional Teams #1 and #6,. The histograms plot the delta of the ordinal rank of student performance from the grade book between paired partners within an Event versus the frequency of that delta pairing for the 5 years of data of SO. The left-most histogram is plotted using all the pairing deltas from the State teams. The middle histogram is plotted using the pairing deltas only for the Regional Team #1's, and the left-most histogram shows the pairing deltas from the 6th-tiered Regional teams.

Figure 31: Equal Pairing of Event Partners on Single Teams



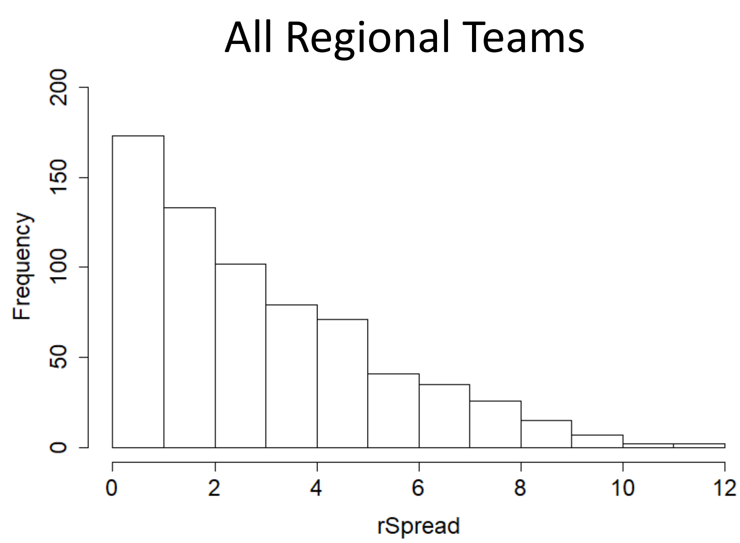
From the graphs, it can be seen that the algorithm did a good job of keeping student pairs reasonably close in ability based on their ordinal ranks within their Events. For Team 1 (middle graph), approximately ½ of the student pairs were within 2 ordinal ranks of each other. For the

State team (left-most graph), ½ of the pairs were within 1 ordinal rank with nearly 2/3 of the pairs within 2 ordinal ranks.

Recall that the Regional teams are composed from the student pool without replacement, so as each successive team is chosen, there are that many more students missing from the pool which can leave gaps in the ordinal ranks within each Event. Also, as the student pool becomes smaller, there is less flexibility for the algorithm to piece together a team. For these reasons, as the lower tiered teams are selected, the average ordinal delta between paired students will tend to become larger. For example, if the pool size starts with 90 students, when the 6th team is composed, the algorithm has no option but to fit the final 15 students on that team regardless of where their ordinal deltas may lay. We see this effect in the right-most graph of Figure 31:, where the graph is much flatter and approaching a uniform distribution.

While the State Team Selection did not have an explicit goal of closely paired student abilities, the graph shows that this was actually a stronger result than for Team 1 of the Regional teams. This result is due less to the different performance weighting of the State (skewed weights) causing pairs to be more closely matched in ordinal rank and more due to having a smaller student pool. Typically before the Regional team selection, the student pools have ranged in size from 150 students (in 2015) down to 95 students (in 2020). Regional team selection cuts the pool to 90 students. When the State team is selected, as much as half the pool of students opts out of being considered for the State team. Thus the single 15-member State team is selected from a pool of 45 to 75 students, resulting in a smaller range of ordinal ranks, and thus closer pairings[143].

Figure 32: Pairing of Event Partners on All Regional Teams



[143] For Regional Team Selection, there are typically 14 students per Event, giving a maximum ordinal delta of 13 between the top-ranked student and bottom-ranked student in an Event. For State, with 60 students in the pool roughly evenly distributed across all Events, there are typically 8 students per Event, giving a maximum ordinal delta of 7 between the top and bottom student in an Event. In a completely random pairing of students, in an Event with 14 students, only ¼ of the students would be paired within two ordinal ranks of each other, whereas in an Event with 8 students, nearly ½ of the pairs would be between students within 2 ordinal ranks of each other. Thus, with a smaller student pool, student pairings will inevitably be more closely paired in ordinal rank when using the same fundamental algorithm regardless of the weightings.

In Figure 32:, the Regional teams' pairing deltas for each school which used Team Selection for establishing all their Regional teams is tallied. For this composite of all Regional teams, the tail has a more linear descent than the State Teams and Team 1's, but still approximately 1/3 of the pairs are within 2 ordinal ranks of each other. This is as expected, since this plot is simply adding together the plots of Teams 1 through Teams 6, where the plots for each of those teams are progressive interpolations of the Teams 1 and Teams 6 plots.

Table 41: tallies the histograms in Figure 31: and Figure 32: to calculate the number and percentage of student pairings which are within *n* ranks of each other by steps of two. For all Regional Teams, over 40% are matched within 2 ranks, and over 2/3's of all pairings are within 4 ranks of each other. This is a good pairing result: since the teams are selected sequentially without replacement, then as the later teams are formed, the pool is getting smaller leaving more limited pairing possibilities. This can be seen in comparing the pairing results for Team 1 vs. Team 6, where Team 1 has 52% of its pairs within two ordinal ranks and 20% 5+ ranks apart, but Team 6 is less well matched with 38% within 2 and 40% 5+ ranks apart. Even though Team 6 has worse pairing, it is not that bad a result, especially considering that the student pool is at its most restricted by the time Team 6 is formed.

Table 41: Ordinal Deltas in Student Pairs

Ordinal Delta	**2**	**4**	**6**	**8**	**10**	**12**
All Regional Teams	310	180	120	70	35	4
All Percentage	43.1%	25.0%	16.7%	9.7%	4.9%	0.6%
Team 1 Pairs	73	41	16	10	1	0
Team 1 Percentage	51.8%	29.1%	11.3%	7.1%	0.7%	0.0%
Team 6 Pairs	29	17	13	10	6	2
Team 6 Percentage	37.7%	22.1%	16.9%	13.0%	7.8%	2.6%
State Pairs	105	26	10	3	2	0
State Percentage	71.9%	17.8%	6.8%	2.1%	1.4%	0.0%

The State Team has very good pairing results due to the smaller pool size and, because it is the first (and only) team formed from the pool, there are no gaps in the ordinal ranks in any Event that were made by students being removed from the pool and placed on an earlier team.

4.2.8 Traditional Method Comparison

In the original method for choosing a State team, the entire set of rules was laid out in this ambiguous sentence:

> *"Usually at least the top 6 students are chosen for the State Team, with the remaining students chosen as necessary to fill in the open Events for the team."*

There is not a single part of this sentence that is not open to interpretation:

- In an 'unusual' year, more or fewer than the top 6 students can be placed on the Team.
- How is Top 6 determined? Regional medals, coach recommendations, number of Events a student is in, and bonus points (awarded solely on determination of the head coach) were some of the different components of the calculations used to rank students, with different components being given very different weights from year to year.

- How are open Events filled? Must the student have studied the Event this year, or does experience in the Event from a prior year count? Is it acceptable to assign a student who hasn't ever been in the Event?

The answers differed from year to year and the answers were never established until all the data was in at which time the results could be easily manipulated by the head coaches by changing the parameters until a more "desired" list of students for the State Team was arrived at.

Table 42: Arbitrary Ranking Example

(A) Score	(A) SID	(B) Score	(B) SID	(C) Score	(C) SID	(D) Score	(D) SID	(E) Score	(E) SID
63	1	73	1	83	1	71	1	126	1
60	2	60	2	80	2	68	2	78	6
60	3	60	3	78	4	66	4	76	9
58	4	60	14	75	6	64	3	75	14
56	5	58	4	74	7	63	6	75	22
55	6	56	5	73	18	62	7	69	11
54	7	55	6	73	9	61	9	68	2
54	8	54	7	70	3	60	5	66	4
53	9	54	8	68	13	56	13	65	15
52	10	53	9	66	5	55	18	64	3
50	11	52	10	60	25	54	11	64	25
50	12	50	11	60	11	54	12	63	17
						54	8	62	7

(A) 3 Event Rank: Sum Best 3 of 4 Events

(B) 4 Event Rank: Sum 4 of 4 Events

(C) Coach @ 10: Sum Coach*10 Best 3 of 4 Events

(D) Coach @ 4: Sum Coach*4 Best 3 of 4

(E) Coach*10 + Best 3 of 4 Events + Prior*15

An example of this manipulative potential is illustrated in Table 42: where there are 5 different equations for calculating the score of the students that will be used for ranking them and determining the top students to include on the State Team. This data is taken from the actual student and coach data from the 2015 season. Equations (A) and (B) are 'objective' in that they include student performance at the Regional competition only. Equations (C), (D), and (E) were independently proposed by each of the three head coaches that year. Each of the 90 students' scores was calculated from the same raw data for each of the 5 ranking mechanisms, and each sub-table displays the top 12 students by their calculated score in order from highest to lowest. The Student IDs (SIDs) were assigned based on Equation (A) and the SID remained the same for the other sub-tables, such that SID 6 in Equation (A) is the same student as SID 6 in Equation (E). The rankings have varying degrees of resemblance; however (A) has very little resemblance to (E) despite coming from the same raw data. The color-coding shows changes in the ranking roster versus the roster to the immediate left, which serves to highlight the degree of change due to the equation and/or parameters. The key to the colors are:

- Tied scores are shown with green cell background
- Red font is a new student who wasn't in the prior sub-table
- Blue font is a student who is in the prior but has changed rank

- Black font is a student who did not change in rank from the prior table
- Purple font is the student who retained top rank in all sub-tables (and the only student to be ranked in the top 6 in all sub-tables)

Only one student ranked in the top 6 students in all 5 rankings.

It's easy to see how this ambiguous 'rule' leads to conflict and consternation over *any* State team, no matter what its composition.

Another factor in hand-selecting the State Team according to these types of rankings is Event coverage. When the students are ranked in this fashion, it is common that students participating in a high performing Event (one in which most teams placed high at the Regional competition) tend to appear high in the rankings. Thus, when a team is made from the top-ranking students, it is common that more than 2 students from that Event will be placed on the team. Given that only 2 students may actually compete in the Event, those other students in that Event cannot compete in that Event. Due to the limitation on the number of students on the team, this also means that one or more other Events will end up with solo students or no students trained in those other Events on the team. In Table 23:, there are 7 Events where all the teams placed in the top 20 places and the average placement for the Event is under 11. Using straight rankings as the criteria for selecting the team would yield too many students from these 7 Events being on the team, and 3 or 4 Events with no trained students on the team, which can leave the team at a disadvantage moving onto the State competition. Note as shown in Table 47:, however, that the ranked teams did not perform significantly different from the algorithm teams. This is in part because the Events which had no trained students in them were also those Events in which the students were not well-trained, so those Events were likely to be on the lower end of the performance scale at State whether or not the students were newly moved into the Event or had trained in the Event over the course of the season.

For the 2014 season, it is possible to compare the potential scores of the different selection methodologies that were used up to that point in time. These methods are:

- Highest Scoring Team at Regional advances to State without any changes[144]
- "Top Students" advance to State regardless of Event filling[145]
- The Hand-selected Greedy Team (constrained)[146]
- Unconstrained Greedy Team[147]
- IP Full Team[148]
- IP Unconstrained[149]

---

[144] This method is generally not considered a "fair" method since if a student is not assigned to Team 1 at the start of the year, that student has no opportunity to go to State no matter how well they do at Regional.

[145] As mentioned earlier, "top" students come from Events with good coaches, thus there are too many students from some Events and zero students from Events with poor coaches, thus students compete in some Events where they have no training or experience.

[146] In 2014, the hand-computed greedy team was composed in fewer than 48 hours wall clock time and was constrained not to include any student below a rank of 25.

[147] This method is the hand-computed greedy team composed without any rank constraint.

[148] For this method, the latest version of the Team Selection IP was run on the old data from 2015 to force-fill a complete team.

[149] For this method, the latest version of the Team Selection IP was run on the 2015 data to create the highest scoring team without requiring that all Events be full.

Using the agreed upon ranking scores of the students, these four methodologies of team selection yielded these team scores:

Table 43: Team Selection Methodology Score Comparisons 2015

Team Methodology	Scoring Potential	Improvement	Unfilled Events	Partial Events
Best Regional Team	Baseline (605 pts)	0%	0	0
"Top Students"	587	-3.0%	4	6
Hand-selected Greedy	645	6.6%	2	2
Unconstrained Hand-selected Greedy	668 [150]	9.6%	3	3
IP Full Team	650	7.4%	0	0
IP Unconstrained	704	16.4%	1	3

In Table 43: the team scores are shown for comparison, where higher scores are better. The points for each team are summed from the Regional score awarded to each student for their ordinal placing in each of their Events, where first place is worth 20 points and 20th place (or lower) is worth 1 point. The "Top Students" has the lowest score of all the methodologies and actually falls *below* the Best Regional Team Baseline score by 3%. All of the other methods create teams with higher potential scores than the "Top Students" method. Clearly, the IP Unconstrained team has the highest Scoring Potential by 16.4% over the baseline and beats the "Top Students" team by nearly 20%. However, the scores don't take into full account the duplication[151] of students in some Events or the lack of coverage[152] of other Events that occurs with most of these methods. "Top Students"[153] is particularly worrisome, since there are 4 unfilled Events and 6 partial Events, for a total of 10 Events (out of 23 Events) that are at least partially handicapped.

Duplicative students in an Event are handy as alternates in case a student in that Event is not able to compete. But there is also the consideration of morale if a student trained in an Event for many weeks or months and then is not allowed to compete at State in their Event. With lack of coverage considerations, it's not a simple task to account objectively for the potential impact. However, a partial Event is clearly less risky than an unfilled Event. Similarly, a partial Build

[150] The unconstrained greedy team was formed by hand after another 4 hours of spreadsheet calculations using the constrained greedy team as a starting point. Since this method likely did not reach the optimal team, it is likely that the unconstrained greedy team may have been able to achieve an even higher potential score.

[151] 'Duplication' is where more than two students on a team are trained in an Event but those extra/duplicative students cannot compete in that Event.

[152] Lack of coverage for an Event occurs if no students on the team had been trained in the Event in the current school year or fewer students than the maximum allowed in the Event are on the team. However, the team still needs to compete in that Event or suffer a significant point penalty if the team does not compete in the Event.

[153] The "Top Students" team is pure and simply a team that is formed to reward the students who earn the highest and most medals with a field trip to the State competition. The "Top Students" will always be numbered and dominated by students in those Events that have good coaches. It is competitive only by luck and happenstance based on the set of *other* Events that the best performing students happen to have in a particular year. For example, in 2015, on the "Top Students" team, there were 6 students from Boomilever, 5 students from Water Quality, and 5 students from Heredity—which is precisely why there were no students at all from 4 poorly performing Events.

Event is less risky than a partial Study Event which is less risky than a partial Lab Event. The performance of the students who voluntarily take on one or more Events that are missing student(s) is highly dependent on the Event coach. A good coach can bring the students to a high level of performance in under 4 weeks of intense work, whereas a mediocre coach will be unable to prepare the students for the high-level competition at State.

Given a higher scoring team with one or more unfilled Events and a lower scoring team with all Events filled by pre-trained students, which team has better potential to perform well at the State competition? Unless the quality of the coach(es) in the unfilled Event(s) is known to a great degree of certainty, the most *consistent* performance will be from a lower scoring team that has pre-trained students in all Events.

Of all the teams from these methods, I would choose the IP Full Team even though it has a lower potential score than some of the other methods simply because all the Events are filled with trained students. However, I would undoubtedly choose the IP Unconstrained Team if I knew I had a high quality coach in the one unfilled Event and that most of the partial Events were Build Events.

Curiously, 2015 is the only competition year in which it was necessary to constrain the IP Team Selection to force-fill a team[154]. In all the other years' data, the State team filled 'naturally' without having to add the constraints that all Events must be fully-staffed. For the Regional Teams, those years with student pools of 110 or more students, at least the first two Regional Teams would fill naturally. In the last two years, with the decreased student pool and an increase in the number of students opting for fewer than 3 Events, it has been necessary to force-fill the first two Regional teams. The trade-off is that the team has a slightly lower overall team potential score, but all Events are represented with trained students.

4.2.8.1 Algorithm or Hand-selected Regional Teams: Which Performs Better?

It is difficult to measure whether or not the competitive team selection process was beneficial, neutral, or detrimental to the competitive teams' performance.

This is because the only deterministic way to measure a team's performance is by having that team perform in a competition against other teams or perform a baseline (standardized) competition. However, in one school year, a CVMS team only competed in one competition, and you can't reconstitute a team using a different formulation and have them then participate in that same competition. Also, since the Events actually included in the competition change from one year to the next, it is not possible to administer a standardized test from year to year.

From year to year, it is simply not possible to make an apples-to-apples comparison, as there are significant differences in the competition from one year to another. Some differences are:

1. Each year, 6 to 9 of the 23 Events are rotated out and new Events come in. Several of the Events that do stay in rotation change in their content or rules[155].

[154] Recall that in 2015 neither the Greedy algorithm software nor the IP software was available, so teams had to be made by hand.

[155] Some (non-permanent) Events stay in rotation for three years before being rotated out. Each year, the rules are made more challenging as coaches and students become familiar with the Event and gain mastery in it. Permanent Events remain every year, but rotate their content areas every year through a 3 or 4 year rotation cycle so that a student will have moved on to high school before a content area is repeated.

2. Different students are in the SOP (last year's 8th graders are gone, and new 7th graders have entered the school).
3. There is a different set of coaches than the prior year. Some of the coaches carry over from one year to the next, but 40% to 60% of the coaches are new each year.
4. The schools in the competition (and their team members) are different. Some schools become more competitive, and some schools become less competitive.
5. There is a different group of Event Captains (the people who make and administer the tests), thus the focus of the Event test and the level of difficulty of the test are different.
6. The pool of students in CVMS SO grew from 75 in 2009 to 150 in 2015.
7. PTMS opened in 2016 and drained half the student body from CVMS over two years, proportionally shrinking the pool of SO students down to 85 Regional competitors by 2020.
8. Competition Day Vagaries. As with any competition, the participants may have an 'on' day or an 'off' day, their device may be accidentally damaged (or purposefully sabotaged) while impounded with SO officials, a student may be ill, the weather may affect performance, an entire Event may be disqualified, and so forth. Any number of things can influence the competition day performances that are beyond the control of the students.

The number of variables introduced by these factors, and the variance due to these differences, make it difficult to determine how the competitive team's performance from one year to the next was impacted by the team selection methodology versus other factors.

Table 44: shows the medal count earned by the 6 teams competing at the Regional competition. The columns numbered 1-20 tally the total number of medals earned by all 6 teams across all 23 Events for that place number (column 1 is first place, column 2 is second, and so on). "Total" shows the total awards earned by all 6 teams across all 23 Events, "Top 10" shows the total medals earned in places 1 through 10, and "Top 5" tallies the medals earned in places 1 to 5. "Pts" is the total team points earned by the top-performing of the 6 CVMS teams, where a higher point total is better[156]. The green highlighted row represents the teams that were selected using the new institutional rules and optimization algorithm[157].

Most notable in the data in Table 44: is that while the total number of medals earned in the years from 2012 to 2016 remains relatively consistent, the number of medals in the Top 5 places has improved by nearly 30% for the 2015 teams and by about 10% in 2016 despite losing ~20% of the CVMS SO student participants to PTMS. It would appear that the optimization algorithm's penchant for pairing students of similar ability level results in more top-end medals: nearly half of the available 1st place, and 43% of the available 1st, 2nd, and 3rd place medals were awarded to CVMS students in 2015. Additionally, for 2020, the CVMS and PTMS teams, which have split the equivalent of the 2015 student body between the two schools, earned 41 Top 5 places and 67 Top 10 places between the two schools, effectively topping even the 2015 year performance.

[156] For Regional, team points in each Event are determined by the equation:

$$22 - \text{place} = \text{points}$$

where first place is worth 21 points and 20th place is worth 2 points. Any place below 20th is assigned 1 point. Total up a team's points across all 23 Events, and that yields a team's total points. The team with the highest point total takes first place. Only the highest scoring team from each school is used to determine overall school place.

[157] 2014 is not highlighted in green, as the Regional teams that year were selected by hand and/or with the VB program.

Table 44: Regional Team Medal Counts

Place	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	Total	Top 10	Top 5	Pts
2020pt	3	3	3	1	7	3	7	3	1	1	2	3	1	2	3	1	0	0	0	1	45	32	17	192
2020cv	9	3	4	7	1	2	1	2	3	3	1	3	1	3	1	1	0	1	1	0	47	35	24	207
2019pt	2	3	1	2	2	5	3	2	3	5	1	2	5	4	0	1	4	4	1	2	52	28	10	289
2019cv	7	5	4	2	3	3	4	1	2	4	2	5	1	1	6	3	2	4	5	2	66	35	21	256
2018pt	4	2	4	4	2	4	3	4	3	2	1	3	3	3	1	6	3	6	1	5	64	32	16	346
2018cv	4	5	6	4	2	4	6	3	5	4	3	3	6	4	1	3	3	4	2	3	75	43	21	241[158]
2017pt	2	7	2	1	3	4	2	7	1	1	1	2	4	2	2	1	5	3	3	2	55	30	15	165
2017cv	5	6	8	9	4	2	4	4	5	1	2	3	6	6	1	2	4	3	3	2	80	48	32	313
2016pt[159]	0	0	0	0	0	0	1	0	1	0	2	0	3	1	2	2	4	0	1	1	18	2	0	60
2016cv	10	7	6	8	6	7	4	7	3	7	3	6	3	2	4	4	4	2	3	1	97	65	37	333
2015	11	10	9	6	6	6	3	4	1	3	2	6	6	4	2	5	2	4	4	2	96	59	42	287
2014	10	7	8	5	3	8	6	8	7	6	3	3	5	3	0	4	6	1	2	4	99	68	33	302
2013	9	5	5	5	10	7	5	7	4	2	5	3	5	4	3	7	5	3	1	3	98	59	34	272
2012	5	5	6	9	5	2	7	7	7	3	6	3	3	1	4	4	5	3	5	5	95	56	30	324
2011	4	8	7	4	9	4	4	2	2	4	5	5	5	1	2	5	3	3	2	1	80	48	32	246
2010	5	4	7	4	5	6	4	2	4	4	5	4	4	5	4	6	1	4	1	2	81	45	31	273
2009[160]	3	3	2	3	3	3	4	1	4	1	6	4	3	1	2	--	--	--	--	--	43	27	14	174

Recall again that teams prior to 2015 were formed by hand at the beginning of the school year based, in part, on prior-year performance of students[158] and, in part, on subjective assessments of those students by the head coaches. The teams were kept static up through the Regional competition regardless of actual performance during the current year. Under the new institution, Regional Teams are formed between Thanksgiving and Winter Break based on reported performance of the students in the current year. From Table 44:, it can be seen that forming teams mid-school year based on current year performance using an objective algorithm produces consistent and significant improvement in team performance over the prior methodology.

4.2.8.2 Algorithm Tiered Regional Teams: Are They Actually Tiered in Performance?

One of the intended goals of Team Selection is to compose teams such that the team with the greatest potential is selected from the remaining student pool. When multiple teams are composed from the student pool, such as is done with the Regional Team Selection, it is possible to compare the performance of those teams at the Regional Competition to see if the teams' performance relative to each other matches their calculated potential. In other words, did the first team selected from the pool outscore the second team selected? Did the second team outscore the third team?

Table 45: and Table 46: show the algorithm Potential Scores for the Regional Teams selected in 2015 and 2020. The teams are listed in each table in order of their Potential Scores from strongest team to least strong team. For the 2015 teams, all teams are fully staffed and competed in all 23 Events. For the 2020 teams, the top two teams are fully staffed and competed in all 23 Events, but the next 4 teams competed in fewer than 23 Events so additional data is included in Table 46: to adjust for the missing Events and make a better comparison between the performance of the teams competing in an unequal number of Events.

[158] While some students do repeat an Event year after year, most students try out different Events than they competed in the prior year. So, prior year performance often isn't the best predictor for future student performance in an unrelated Event. For example, a student doing well in Anatomy last year doesn't necessarily have much bearing on how well that student will do building a balsa wood bridge this year.

Table 45: 2015 Regional Teams Relative Performances

Algorithm Potential Score	Potential Team Order	Regional Team Order	Regional Points[159]
1949	Curiosity	Curiosity	287
1911	Rosetta	Rosetta	251
1750	Dawn	Ulysses	227
1647	Ulysses	Galileo	224
1420	Galileo	Dawn	202
1212	Huygens	Huygens	163

For the 2015 Regional Teams, the Regional results are given in points where higher points are a better result. The table shows that the teams finished in the order predicted by their Potential Score with the exception that Team Dawn (3rd Place amongst the school's teams by Potential Score) finished 5th place relative to the other teams in the table.

For the 2020 Regional Teams, the Regional results are given in ordinal place and points where *lower* points are a better result. The raw Regional Results (by Regional Place and Regional Points) followed exactly the order predicted by the Potential Scores. However, since four of the teams are partial teams, they are awarded maximum points for each Event in which they do not compete, so their Regional Points scores place those teams lower than they should be if they are judged only by the Events in which they actually compete. For the partial teams, the "no-show" points are subtracted from the official Regional Points and these Regional Corrected Points are divided by the number of Events participated in by that team to yield an Average Score Per Event. The relative strengths of full and partial teams can be compared using the Average Score. With this comparison, the finish order of the 2020 teams changes, where Volta now bests Curie and Mendel bests Sagan. Even with this comparison method, the ordering of the teams' performance closely follows the 'tiering' order of the Potential Score.

Table 46: 2020 Regional Teams Relative Performances

Algorithm Potential Score	Per Event Score	Team Name	Regional Place	Regional Points	# Events	Regional Corrected Points	Average Score Per Event
2020.0	87.8	Oppenheimer	4	207	23	207	9.0
1694.9	73.7	Curie	11	319	23	319	13.9
1435.6	68.4	Volta	12	341	20	230	11.5
1234.9	65.0	Salk	21	448	20	337	16.9
921.2	54.2	Sagan	27	561	18	376	20.9
514.9	39.6	Mendel	35	658	11	214	19.5

It's easy to explain the performance difference in the teams between their Potential Score order and their Regional Points performance. The students are human and can have good days or bad

[159] Between 2015 and 2020, the Regional Team scoring methodology was changed. In 2015, higher scores indicated better performance; by 2020, lower scores indicated better performance. The change was made so that San Diego Regional's scoring was the same as Southern California State and National's scoring system.

days, just like any competitive endeavor will sometimes yield surprising results[160]. Also, the Potential Score is based on student performance and assessment around the end of November, whereas the competition occurs in early February. During the two months between team selection and the Regional competition, students continue to learn and work, and their knowledge base and experience changes. Additionally, as discussed elsewhere, the grade book is not necessarily an accurate assessment of student ability; however, the score from the grade book is what is used to assign students to the tiered teams.

Yes, the tiering of teams works and succeeds in creating a series of teams from greatest potential for performance to least potential performance.

4.2.8.3 Algorithm or Ranked State Team: Which Performs Better?

It is even more difficult to quantify whether or not the State team selection was beneficial, neutral, or detrimental to the competitive State team's performance. All of the differences and variables that are present with comparing the performance of the Regional Team selection process are also present with the State Team selection, further hindered by the severe limitation of having only a single data point each year[161].

Table 47: State Team Medal Count

Place/ Year	1	2	3	4	5	6	7	8	9	10	Total	Top 3	Points	State Place	Method
2020													N/A	N/A	Cancelled
2019	3	1	1	0	2	1	2	1	0	1	12	5	201	4	TSIP
2018	2	2	0	1	1	2	1	2	1	0	12	4	217	5	TSIP
2017	1	1	1	4	2	0	1	3	0	1	14	3	203	4	Greedy
2016	3	2	1	2	1	3	1	0	1	2	16	6	163	4	Greedy
2015	4	3	4	1	0	1	4	1	1	2	21	11	135	2	Greedy
2014	3	6	2	1	1	3	0	1	0	0	17	11	126	2	Manual Greedy
2013	6	0	3	2	4	2	1	0	0	1	19	9	134	2	Ranked
2012	2	3	2	2	4	3	2	0	0	3	21	7	143	3	Ranked
2011	4	2	2	6	0	0	1	3	0	1	19	8	134	2	Ranked
2010	3	4	3	1	1	1	1	1	2	2	19	10	138	2	Ranked
2009[162]	1	0	0	2	3	1	0	1	1	1	10	1	229	5	Student
2008	3	4	1	2	2	1	0	0	0	1	14	8	151	3	Ranked
2007	Unknown # of medals earned												Unk.	2	Ranked
2006	Unknown # of medals earned												Unk.	2	Ranked

[160] For example, Team USA vs. Team CCCP in the 1980 Winter Olympics, where the highly favored Soviet team lost to the underdog US team.

[161] With Regional Team selection, there are 6 teams and 90 students, which is 6 times more data than the State Team selection where there is only 1 team and 15 students.
In addition, there are typically just over 100 teams in each Regional competition, and approximately 30 teams in each State competition.

[162] 2009 was an aberrant year with regard to team performance, because in that year, the State team was selected by the Coordinator's own middle school-aged children. The team score for 2009 was 50% worse than 2008's team and 70% worse than 2010's team.

It is possible to look at the performance of the teams over the years, ignore the confounding factors and just look at raw performance. Did the CVMS State Teams chosen by the algorithm do better, the same, or worse than the teams chosen manually? Table 47: takes a statistical look at the performance of the CVMS State Team over all the years in which it competed at the State competition to help answer this question.

Table 47: shows the medal count earned by the State team competing at the State competition. The columns numbered 1-10 tally the total number of medals earned by the State team across all 23 Events for that place number. "Total" shows the total medals earned in places 1 through 10, and "Points" is the total team points earned by the CVMS State team, where a *lower* point total is better. The rows with green background (2014 and 2015) represent the two State teams that were selected using an optimized team approach[163]. The row with red background (2009) is the aberrant selection year and had 5 Regional teams. The rows with the yellow background (2008 and earlier) also had fewer than 5 full Regional teams. There was no record of the medals and team points earned from the 2007 and 2006 seasons.

Comparing the Total Medals, Top 3 Medals, Points, and State Place at the interface of the changeover from Ranked team selection to algorithmic team selection (seasons 2013, 2014, and 2015), there is almost no difference in the State results between those 3 years. At best, there might be a slight advantage in the number of top 3 medals earned for the algorithm teams and in total medals for the software driven algorithm in 2015. However, there are not enough data points to say this pattern is statistically significant. It is very apparent that the Student-chosen team from 2009 was significantly lower performing than the ranked teams and the algorithm teams. It is also apparent that the competitiveness of the CVMS teams decreased following the opening of PTMS and the subsequent decrease in the CVMS student body: those students were now competing *against* CVMS whereas previously they would have been on CVMS's teams. Given that the PTMS teams perform at roughly the same level as the recent CVMS teams, there is clearly a brain drain impacting the performance of the CVMS teams. The opening of PTMS likely explains the approximately 50% drop in top 3 and top 10 medals earned by CVMS from its peak in 2015 to 2018 and leveling off in 2019.

Thus, it appears that the algorithmic team selection certainly did not diminish the performance of the CVMS State teams, and may have slightly improved the performance of the team. It appears that the decrease in the student pool had a larger negative impact on the team performance than the change in selection methodology. The pool size change impact makes sense assuming the pool was reduced proportionately across all skill levels, thus causing the State team to reach lower into the pool to fill the team.

A final note on Ranked Teams versus optimized teams: while it is great to get the highest performing students on a team regardless of which Event they are assigned to, this always comes at the expense of the lower performing Events. If the goal of Team Selection is to reward the highest performing students, then forming a team through ranking is a great way to go.

[163] The two State teams selected using an optimization on the whole team potential: 2014 being selected by hand implementing an approximate algorithm using only Regional medal count, and 2015 using a computer program and multiple sources of performance information (homework, tests, mini-competition, Regional, and Muscatel Invitational results). These are the two years that are most similar to the years between 2010 and 2013.

However, if the goal of Team Selection is to form the most competitive team with the highest potential to score well in a competition and advance to the next level of competition, then ranking is a very poor way to select a team. Ranking wastes talent by bringing duplicative/surplus students from high scoring Events who can't compete in those Events because there are too many of those students, and it does this by neglecting the lower performing Events by not bringing any trained students from those Events on to the team (and hoping that the surplus students can do well in Events with which they are not familiar). While it is not possible to do better than 1st place in any Event regardless of the number of students talented in that Event who are on the team and who are not allowed to compete in that Event because there are too many of them, it is possible to have the neglected Events finish dead last due to unskilled (or uninterested) students and remove any chance of advancing to the next level of the tournament. If the goal is to advance in the competition, it is a far superior strategy to target to surrender 1st place in some Events in exchange for finishing much higher than last place in other Events. That is, a loss of one or two places in a strong Event which allows the gain of 10 or more places in a weak Event nets a significant gain in the overall team score.

4.3 Crowdsource Tool

The *concept* of overall team performance optimization, aka "Moneyball", was easily understood and accepted by the parental demographic. Distilling the concept into a series of algorithmic steps that would accomplish the end goal was beyond the grasp of most parents—even those with technical and analytic backgrounds. Converting the concept of performance optimization into a functional algorithm that would run automatically without human intervention likely decreased parental discontent with the State Team Selection process. An automated optimization algorithm eliminated the specter of human bias in making choices of this student over that student. It also removed most of the opportunity for smoke-filled backroom deals where the head coaches and their adult friends were thought to work out deals to get their kids on the State team regardless of merit[164]. The transparency provided by the new SOP institutional structure and its facilities were key to this change.

The single greatest contributor to tamping down parental discontent came from the idea of Crowdsourcing the State Team selection. The Crowdsource Tool acted as an educational tool, which enabled participants to understand the complexity and combinatorial nature of team selection essentially through a game/puzzle/challenge rather than reading a detailed, boring mathematical description of how team selection worked. The game-like nature of the tool was further enhanced over the first 3 years of the program because it really was a race between the algorithm (which took 4 days to get close to an optimal solution) and humans to see who could post the best solution before the close of the contest window.[165] The Crowdsource Tool also provided a huge window of transparency into the Team Selection mechanism.

[164] In the first two years I was involved with SO, 10 to 20% of the families in the SOP would complain to the school principal about the 'unethical' backroom behavior of the head coaches regarding State Team Selection. In the first two years of operation within the new institution, there were zero complaints that went to the principal regarding State Team Selection.

[165] The first year, the best solution was posted by a parent 2 hours before the algorithm found the identical solution, and the second year, the humans found the solution an hour after the algorithm had found it (but the algorithm's final solution was not posted until the contest window closed). In subsequent seasons, the algorithm has always finished first since the student pool size has decreased which significantly decreases the search space for the greedy

The Crowdsource Tool educated the stakeholders (particularly the parents) on exactly how difficult picking a high potential team is while constrained by which students have which Events, their performances in those Events, the competition rules of 15 students maximum, and a requirement for full Event coverage. The parents learned that the simplistic approach of selecting the "top" one or two students from each Event yields a team with more than 15 students, which violates the competition rules for team size. The parents discovered that it may be easy to pick the first 8-12 students on the State team, but picking those last 3-7 students to fill out the remainder of the team is extremely difficult and often requires the removal and replacement of some of their initial picks in order to compose the team optimally. Everyone quickly learned how difficult this 'simple' task is. The tool also makes it simple and quick for a parent to build a team around their child, so the parent can often quickly discover that the highest scoring team *cannot* be a team that incorporates their child.

In the seasons in which a school used the Crowdsource tool for State Team selection, the number of complaints about the final State Team was typically under 2 and often zero. In the seasons a school did not use the Crowdsource tool, there were between 5 and 10 parents who questioned the results—which was still well under the number of complaints under the old regimen. In one season, a school posted the optimal team as the initial seed team, which only succeeded in frustrating the participants since they could not find a better team, so activity on the tool ceased within the first 24 hours of the 4 day contest window as word spread that it couldn't be beaten and everyone gave up. In another season, one school chose not to use the tool, and had a particularly virulent response from one parent. This resulted in intervention from the principal and the SOP setting up a private hosting of the tool for that one parent. During the two hours that the parent played with the website, the parent could not form a team with their child on it such that the team score was higher than the algorithm's team. The parent conceded that the chosen team was indeed the best team to move forward.

4.4 Institutional Structure

When I started this project, I was certain that the Team Selection algorithm would be the be-all and end-all key to smoothing out the contention amongst parents in the SOP. Very quickly, it became apparent that Event Assignment was just as important as Team Selection. It was also painfully apparent that an institutional structure was needed or none of the algorithms would be able to make any positive difference. Thus, we started the project by creating an institutional structure that would survive from year to year and have institutional memory via written documents and people on a steering committee who stay with the SOP for more than one year.

Teams are no longer formed until around Thanksgiving, which has resulted in a decrease in cut-throat competition amongst the students (and parents) and fosters more cooperative behavior. This also serves as an incentive for students to perform well in the current year rather than resting on their laurels of last year's performance. The grade book ensures that students and parents have the opportunity to track their progress.

Thus, the second largest contributor to decreasing parental discord was the imposition of structure and consistency on the SOP in the form of an institution. Critical to this was the inclusion of written rules and procedures which were provided to all participants at the start of each competition year. Obviously over the course of the competition year, the institution needed

algorithm, and even more so once the IP algorithm was implemented and optimal solution times decreased to less than 1 second.

to be respected and the rules and procedures followed without making up creative interpretations of them or bending them to suit certain individuals' needs. In other words, no fair changing the rules of the game after the game has started—and people would certainly cry "foul" if the rules were changed at the end of the game when the outcome is known.

As with all institutions, if the leaders of the institution are unwilling to work within the framework of the institution, and the people who are supposed to provide checks and balances on those leaders are unwilling to exercise their oversight authority, then the institution will cease to function and perhaps cease to exist.

As such, certain personalities in a leadership position can easily distort, pervert, or destroy an institution[166]. Based on numerous interactions over the course of each competition year with a total of twenty-one head coaches at five schools over the six years of this project, seven of those head coaches blatantly manipulated Events, data, and/or interpretation of rules and procedures to provide advantage to their own children (in the case of 3 out of 15 of the adult head coaches) or themselves (in the case of 5 out of 6 chairpersons of the high school committees). Two of the adults were removed from their position by their respective school's SO steering committee, and the third adult served 1½ terms as head coach with a steering committee composed of compliant members whose children were also given favorable treatment. The only consequences[167] for the high school chairpersons' transgressions were that several of the rank and file SO participants would resign each year from their team in protest and discontinue participating in SO for the remainder of their time in high school.

The good news is that for those two schools' SOP, the institutions fully recovered following the departure of the head coaches who were violating the spirit and rules of the institution.

It is key that the Steering Committee not be composed of long-time personal friends of the Head Coach. As laid out in the institutional document, the Steering Committee needs to be composed of adults from various backgrounds, who have students at different grade levels, and a few who have had prior experience with SO. The Steering Committee must also be willing to redirect or remove a Head Coach who does not follow the spirit of the school's SO institution or the spirit of the SO competition. In the face of a manipulative Head Coach, without a strong Steering Committee, the rules, procedures, and algorithms are meaningless.

Finally, there is a tremendous savings in time and labor through using a website and database for registration, gradebook, contact lists, Event assignment, team selection, and other SOTM administrative tasks. What used to be a full-time occupation for a non-working parent for an 8 month period can now be managed by a working parent on the side. Large, contentious tasks that used to take a week for two or three parents now take an hour with one parent. Complaints and questions that used to come in at a rate of 10's in a normal week and in the 100's during Event assignment and State Team Selection are have been reduced by more than an order of magnitude due to written rules, systems transparency, and access to the website to look up information about Events and student scores.

[166] These paragraphs certainly sound like they could be a commentary on current Events in the United States…

[167] The principal did recognize and acknowledge there was a problem with the cliquishness of the high school SOP, especially with regard to how the committee members received special treatment. However, the principal declined to follow up with the students on the committee or take any action of any kind. The high schools' SOPs continue to operate with little adult supervision and no leadership training.

The school principals are happier too, as the number of parental complaints that they have to field regarding SO have decreased 100 fold since the start of this project.

5 Conclusion

This project set out to bring order, logic, and reason to an SOP that was in chaos. We were able to eliminate the chaos with the creation of a predictable and reasonably unambiguous institutional structure with written rules and procedures. We were able to eliminate most of the parental in-fighting by introducing objective, performance based tools which inhibit the more undesirable aspects of human bias within the SO domain. Sociologically, the institution design was the most important factor in the seven year success of the SOP revamp. The upfront rules and transparency of the systems combined with input from multiple sources on the performance of the students built confidence in the participants that the system is 'fair'.

Algorithms are more defensible if they are seen as either giving people more of what they want or promoting performance-based outcomes or both. It is better if the algorithms are easily explainable. Even if they are very complex algorithms, if they can be demonstrated in an engaging way (such as the Crowdsource page for Team Selection) or if they can be explained by an analogy reference (such as "Moneyball"), then it is much easier for people to agree to the methodology upfront.

5.1 Event Assignment

The design goals of Event Assignment were met:

- Do a better, more efficient, and equitable job of giving students the Events they want.
- Distribute students uniformly across Events so as to not bind team selection with low enrollment in some Events and bar students from competing with high enrollment in other Events.
- Build confidence of the participants in the system and algorithms through a good experience with a high-quality (low Misery Index) result with their Event Assignment.

In addition, Event Assignment using a multi-iteration, multi-round RSD algorithm for SO has these characteristics:

- Stable. No student, A, should be willing to trade one of their assigned Events, X, for an Event, Y, from any student, B, where B values X more than Y and A values Y more than X.
- Pareto Optimal. No student can be made better off without another student becoming worse off.
- Relatively Game-proof. All students were assigned their 1st available choice Event, families who attempted to game the system were more likely to be disappointed with their results.
- Time & Labor Saving. RSD completes typically in less than 5 minutes vs. one week when performed manually or with the VB script.

The Event Assignment algorithm consistently provides superior results in significantly less time than other methods, especially as the scale of the number of students (or courses) increases.

5.2 Team Selection

The design goals of Team Selection were met:

- Teams are created based on student performance rather than opinion.
- Teams are fully staffed in all Events without need for re-training.
- Teams are configured to be strongest across all Events rather than being strong in specific Events at the expense of being weaker in other Events.
- Team score is predictive of *relative* team performance, thus tiering teams works.
- Students of equal ability tend to be paired together for Regional teams.
- Students are able to compete at Regional in the Events in which they studied.
- Crowdsourcing educates and challenges stakeholders with regard to Team Selection and makes the Team Selection process transparent.

In addition, Team Selection based on student performance has these characteristics:

- Fully Staffed. Top teams are fully staffed with trained students in all Events. While teams which are fully staffed with trained students in all Events won't necessarily perform better than teams with some Events that have untrained students, those teams do perform more consistently. Additionally, those teams require much less preparation and effort from the Event coaches and students to prepare for the State competition.
- Pairing Parity. Students of approximately equal ability tend to perform better than pairs of students who are widely mismatched, and the matched pairs tend to perform as predicted by their scores. Matched pairs also lessen the issues associated with Free Riders.
- Tiered Performance. The team scores are good predictors of *relative* team performance within a school's pool of teams. The top scoring team from a school sets the school trophy at competitions and determines whether a school advances to the next level of competition or not. Hence it is important to form as strong of a team as possible, with a second team also as strong as possible a backstop if the top team chokes so that a school maximizes its probability of advancing in the SO competitions.
- Team Strength. The algorithm teams did not perform appreciably better than the hand-picked (top student) teams, however, the quality of their medals improved. And the teams remained competitive over the years despite the student pool having decreased to half of the peak size.
- Time & Labor Savings. Converting team selection from pencil, paper, and spreadsheets to a computer program saves a tremendous amount of time and labor. A transparent, objective algorithm doing the team selection removes much of the consternation from parents that is a natural product of opaque team selection process done by humans behind closed doors.

The Team Selection Algorithm produces optimal numeric results for team potential performance. The results are arrived at mathematically, minimizing human bias. Obviously, there is still human bias and error in the grading of student performance by Event coaches. There is also human bias in the parameter selection for all aspects the SOP. This is why parameter selections and scoring rubrics should be selected and locked in well in advance of knowing any potential outcomes. As with Event Assignment, algorithmic Team Selection results are mathematically superior to other methods, and are substantially more so as the scale of the data increases.

It also must be remembered that human beings are the ones being evaluated and paired for placement on teams, not unemotional "agents" or "econs".[35] It is difficult to incorporate personal relationships in any sort of objective and fair manner in an algorithm, even more difficult with adolescents. While citizenship (the behavioral kind) is included in the performance assessment of the students, the algorithm does not know and does not care whether student A gets along with student B. So the algorithm cannot assess if such a pairing might be super-additive or destructively sub-additive. Fortunately, so far, the potential pairings of students with strong personality conflicts has been avoided, whether by pure coincidence or other effect of the system. That is to say, no head coach using this system has had to make a judgement call to eliminate a student from the team selection process due to personality conflicts.

5.3 Crowdsource Tool

The Crowdsource Tool brought an important element of transparency to Team Selection. It is also a great teaching, learning, and do-it-yourself participation tool. The tool was engaging because of its puzzle-like challenge, which made it that much more effective as an educational tool and as a tool that brought further transparency to the contentious topic of Team Selection. The Crowdsource Tool is responsible for significantly decreasing the level of confrontation over Team Selection as clearly demonstrated by the difference in noise level from parents when the Crowdsource Tool was used in conjunction with Team Selection vs. when Team Selection was done without participant access to the Crowdsource Tool.

5.4 Institutional Structure

Without an institutional structure providing a consistent framework, rules, and procedures within which to run the SOP, there is little doubt that this entire project would have collapsed before the end of the first year of operation. Informing people upfront about the rules, procedures, and sequence of activities over the course of the year sets expectations for the participants. Following through as promised builds the confidence of people in the institution and they are more inclined to put stock in the fairness of the system, especially when they have a good experience at the start of the year with Event Assignment.

The key features of the institution turned out to be:

- A Steering Committee to assist the Head Coach with running the SOP as well as serving as a check and balance on the Head Coach. An unchecked, selfishly-motivated Head Coach *can and will* ignore all aspects of the institution to seek as much personal advantage as they can garner.
- Crowdsource Team Selection.
- A grievance or dispute handling procedure in place and making people aware of the procedure upfront. Simply knowing that a dispute can be filed and will be handled builds trust in the institution and appears to decrease the number of disputes that are brought up.
- An easily accessible grade book saves argument. Students/parents know where their kids are on the totem pole in each Event and can check the grade book at any time. Providing Event Coaches with control over their rubric and data entry provides the Event Coaches with a sense of empowerment.
- The non-human intervention aspect of algorithms for performing Event Assignment and Team Selection is more important than the quality of the algorithms. As long as a

> methodology is reasonably logical and fairly easy to understand conceptually, then the algorithms' ultimate efficiency is not critical: close enough counts.

As a final note on the institution, vigilance and oversight by the Steering Committee is critical to the continued fair operation of the SOP. Unchecked Head Coaches will pervert the institution to their advantage. In the case of parent-led programs, Head Coaches are tempted to give advantage to their own children, the children of their friends and/or the children of collaborating adults. In the case of student-led programs, the student committees unflinchingly give advantage to the committee members and then their close friends. With a Steering Committee of a reasonable size, it is more difficult for a Head Coach to co-opt the Steering Committee members to assist with their manipulation of the system.

APPENDICES

A. Structure of a Science Olympiad Team

An SO team is composed according to the following requirements:

- A maximum of 15 students
- Competing in a maximum of 23 Events
- Most Events are limited to 2 or fewer students competing from a team, a few Events (usually 1 or 2) are limited to 3 or fewer students competing from a team

These requirements are somewhat arcane in that the numbers that define the team and competition Events are not integer multiples of each other, which complicate the association between students and Events. For example, a simple integer relationship between team size and Event count could be 15 students and 15 Events, where all 15 students compete in two Events, with every Event using a pair of students.

With these more arcane rules, in a typical competition year, 22 Events allow up to two students per team to compete and 1 Event allows up to 3 students per team to compete, leaving up to 47 student-slots per team. The math works out as:

22 Events * 2 student-slots/Event + 1 Event * 3 student-slots/Event = 47 student-slots

for a fully-staffed team. Some years may have up to 3 Events with 3 student-slots, yielding a maximum of 49 students-slots for a full team.

On a typical team of 15 students, there can be 13 students competing in 3 Events and 2 students competing in 4 Events, thus filling all 47 student-slots on the team. There is no restriction on fielding a team that competes in all 23 Events using fewer than 15 students (meaning more students must compete in 4 or more Events, for example, 12 students each compete in 4 Events). There is no requirement that a student compete in more than 1 Event.

There is no restriction on fielding a partial team at the competitions. A partial team is a team with one or more of the following characteristics:

- fewer than 15 members
- does not compete in all 23 Events
- competes in one or more Events with fewer than the maximum number of allowed students (fewer than a pair or triple of students)

At the competitions, there are typically 5 or 6 testing blocks, which sets a theoretical limit that a student cannot compete in more than 6 Events. The testing blocks are typically 50 minutes long with a 10 minute transition time between blocks. If a student were to compete in all 6 blocks, there would be little opportunity for eating or snacking over the 6 hour testing time, plus that is a long period of time for a middle school or elementary school student to be in continuous testing. As a practical matter, then, students won't compete in more than 5 Events and the vast majority of students compete in 4 or fewer Events. It is technically possible, however, for a student to compete in more than 6 Events as there are usually 3 or more 'build' Events which are tested outside of the regularly scheduled blocks, and it would be possible to test in multiple of these 'build' Events within the testing day and not conflict with a student's Events being tested in the regular block schedule. If this happens, it is extremely rare due to the unusually high demand and pressure on the student to be able to accomplish this feat.

For practical considerations, a school where SO is part of the regular teaching curriculum, students typically don't compete in more than 5 Events; schools where SO is an extra-curricular activity, students typically don't compete in more than 4 Events. However, in schools with SO as part of the curriculum, it is not unusual for students to *study and prepare* in as many as 8 different Events.

B. SOTM Rules & Procedures

The rules and procedures for the CVMS SOP are contained in a series of documents that are provided to all students, parents, coaches, and other stakeholders involved in the CVMS SOP[168]. The documents in the package delivered to the stakeholders include:

1. **1 CVMS SO 2018 Application Instructions.docx** – how to register as a student to participate in the CVMS SOP.
2. **3 Science Olympiad Mission.docx** – The Mission, Vision, Core Beliefs and Values of the National Science Olympiad organization.
3. **4 CVMS SO Rules v15022.pdf** – The institutional structure and guidelines of the CVMS SOP.
4. **6 CVMS SO Pledge.docx** – The parent and student pledge of appropriate behavior and interactions with others while participating in SO.
5. **7 CVMS SO Responsibilities.docx** – The expectations and responsibilities of students and parents participating in CVMS SO.
6. **8 CVMS SO Student Team Selection Guidelines.docx** – a simplified description of the team selection process.
7. **9 Dispute Resolution for CVMS SO.docx** – A description of the dispute resolution process in the CVMS SOP.

[168] These documents have also been provided as templates to other schools for organizing their SOP. These schools have adopted the institution in full or modified parts to fit the particular needs of their schools.

C. Data Sources

Files of Interest:

C:\Users\clifner\Documents\Clifner\SciOly\CVMS-SciOly2014\Admin

- CVMS SciOly TeamSelection.pptx – discussion of manual algorithm implementation, reasoning and considerations.
- BackUpForStateTeamSelection.pptx – additional detail on manual algorithm implementation, etc.
- StudentPrefs.xlsx – student Event preferences and schedule availability from 2014 season. (.\StudentCourseRankings)

Websites:

- https://scioly.org/wiki/index.php/Carmel_Valley_Middle_School - historical performance of CVMS. Must look at edit history to find information, specifically the archive from 17:17, 8 May 2016.
- https://so.augustsystesm.net

Sample Theses:
C:\Users\clifner\Documents\Clifner\Chapman\SampleThesis
    A Computational and Experimental Examination of the FCC Incentive.pdf
    DissertationChecklist_PhD_2019.08.19.pdf

Papers:
C:\Users\clifner\Documents\Clifner\Chapman\Thesis\AssignmentPapers\unread
    EducationAssignmentProblemAndReviewArticle.pdf
    RandomSerialDictatorship-TheOneAndOnly.pdf
    ParetoOptimalMatchStudentsToCourses-Cechlarova.pdf

Results Data:
C:\Users\clifner\Documents\Clifner\SciOly\
    .\CVMS-SciOly2016\TeamSelection
        2016-Regional-Team List.pdf
    .\CVMS-SciOlyResults\State
        2016 State Prelim Scores B.pdf
    .\2019\Results
        2019 Div B State Final Scores.pdf

Docs:
    Chapman_CADS_Thesis_LanceClifner-2020.docx
    Ryan-Chapter1.docx
    CVMS-SciOly-2015.docx
    TeamSelection.v03.docx
    TeamSelectionPairingScoresVsResults.docx
    ClifnerDissertation-2020-04-20.red-line.docx

Excel Files:
    2015/VB/RSD Event Assignment Comparisons
        C:\Users\clifner\Documents\Clifner\Chapman\SciOly
            CVMS-SciOly-EventAssignment-2014+2015.xlsx
        C:\Users\clifner\Documents\Clifner\SciOly\Data
            2015-EventAssignmentStats.xlsx
    SO Data
        C:\Users\clifner\Documents\August\Projects\Assignment\rsd\d
            CVMS-output-2k.csv
            CVMS-output-10k.csv
            CVMS-output-100k.csv

CVMS-output-firstcome.csv
CVMS-CourseData-2018.csv
CVMS-StudentData-2018.csv
(Improvement and differences in Misery Index with iterations and first-come, first-served mechanism)

C:\Users\clifner\Downloads

43-TeamSelectionStudentData-2020-04-09T16_41_25-07_00.xlsx (CVMS State Team Data)
43-AttendanceData-2020-04-09T14_10_18-07_00.xlsx (CVMS State Team Attend & Cit Data 2020)

C:\Users\clifner\Documents\Clifner\SciOly\CVMS-SciOlyResults\Regionals

2020-Regional-Stats-CVMS.xlsx

C:\Chapman\c20

cfo-full2-nodrop4-plain.xlsx (final state team CVMS 2020)

C:\Users\clifner\Documents\Clifner\SciOly\Data\FromBen

NormData.xlsx
SubjectData.xlsx
AllData.xlsx
Summary.xlsx

C:\Users\clifner\Documents\Clifner\SciOly\SOTM-Documentation

2019-12-TeamSelectionExample-Simple.xlsx

C:\Users\clifner\Documents\Clifner\Chapman\SciOly\DataAnalysis

2019-09-14-ResultsSummary.xlsx

From Ben Email:

2016-Regional Data Normalized.xlsx

C:\Users\clifner\Documents\Clifner\SciOly\Data\NewData

RegionalRegressionDataMaster.xlsx
2020CVMS-Regional.xlsx
2019CVMS-Regional.xlsx
2018CVMS-Regional.xlsx
2017CVMS-Regional.xlsx
2019CVMS-State.xlsx
2018CVMS-State.xlsx
2017CVMS-State.xlsx
2020PTMS-Regional.xlsx
2019PTMS-Regional.xlsx
2018PTMS-Regional.xlsx
2017PTMS-Regional.xlsx
2019PTMS-State.xlsx
2018PTMS-State.xlsx
2017PTMS-State.xlsx

2015 State Team Selection Data and Comparisons

C:\Users\clifner\Documents\Clifner\SciOly\CVMS-SciOly2014\Admin\RegionalResults

Placement Matrix-WithTeamSelectionFormatFor-LP.xlsx

AGE Data C:\Users\clifner\Documents\August\Projects\AGE\age\uploads

assignmentOutputOceanAirElementarySchool_6-misery.csv and .xlsx
assignmentOutputOceanAirElementarySchool_6-classfill.csv and .xlsx
assignmentOutputOceanAirElementarySchool_6-studentfill.csv and .xlsx
assignmentOutputOceanAirElementarySchool_6-distribution.csv and .xlsx
assignmentOutputOceanAirElementarySchool_6.xlsx

D. SOTM & Grade book Website

The SOTM website is found at this URL:
https://so.augustsystems.net

The SOTM user manual for Head Coaches is in the document:
SOTM_Description.docx

Instructions for setting up SOTM for a new SO competition year are in the document:
SOTM-Creating a New SO Year.docx

E. References

E.1 Stable Marriage

[1] Gale. D. Shapley. L. S. *College Admissions and the Stability of Marriage*. The American Mathematical Monthly. Vol. 69. No. 1 (Jan, 1962). 9-15.

(Gale-Shapley, Alvin Roth) (https://en.wikipedia.org/wiki/Stable_marriage_problem).
(gale-shapley-college-admissions_stable-marriage.pdf)

[2] Huang. Chien-Chung. *Cheating by Men in the Gale-Shapley Stable Matching Algorithm*. Algorithms – ESA 2006, Lecture Notes in Computer Science. Springer Berlin Heidelberg. **4168**: 418–431. Azar, Yossi. Erlebach, Thomas (eds.).

(marriage_cheating-in-GaleShapley.pdf)

[3] Klaus. Bettina. Klijn. Flip. *Stable Matchings and Preferences of Couples*. Journal of Economic Theory 121 (2005) 75–106.
(StableMatchingsAndPreferencesOfCouples-KlausAndKlijn.pdf)

E.2 The Assignment Problem in the Education Domain, College Class Scheduling (auction and tokens)

[4] Bhushi. K. ed. Farm to Fingers: The Culture and Politics of Food in Contemporary India Cambridge. Cambridge University Press. 2018.

[5] Budish. Eric. *The Combinatorial Assignment Problem: Approximate Competitive Equilibrium from Equal Incomes*. University of Chicago School of Business, October 8, 2009.

(Budish-approxceei-Oct2009.pdf)

[6] Budish. Eric. Cantillon. Estelle. *The Multi-unit Assignment Problem: Theory and Evidence from Course Allocation at Harvard*. November 8, 2009.

(budish-cantilion-course-alloc-harvard-nov2009.pdf)

[7] Budish. Eric. Che. Yeon-Koo. Kojima. Fuhito. Milgrom. Paul. *Designing Random Allocation Mechanisms: Theory and Applications*. American Economic Review 2013. 103(2): 585-623.

> We generalize the theory of randomized assignment to accommodate multi-unit allocations and various real-world constraints, such as group-specific quotas ("controlled choice") in school choice and house allocation, and scheduling and curriculum constraints in course allocation.

(randomassignment.pdf)

[8] Cechlarova. Kataria. Klaus. Bettina. Manlove. Dave. *Pareto Optimal Matchings of Students to Course in the Presence of Prerequisites*. April 24, 2018.

(ParetoOptimalMatchStudentsToCourses.pdf)

[9] Chingos. Mathew. *What Matters Most for College Completion?* AEI. May 30, 2018.

[10] Faudzi. Syakinah. Abdul-Rahman. Syariza. Adb Rahman. Rosshairy. *An Assignment Problem and Its Application in Education Domain: A Review and Potential Path*. Hindawi Advances in Operations Research. Volume 2018. Article ID 8958393.

(EducationAssignmentProblemAndReviewArticle.pdf)

[11] Kadam. Vinod. Yadav. Samir. *Academic Timetable Scheduling: Revisited*. International Journal of Research In Science & Engineering. April/2016. 417 – 423.

(timetabling.pdf)

[12] Olson. Mark. Porter. David. *An Experimental Examination into the Design of Decentralized Methods to Solve the Assignment Problem With and Without Money*. Economic Theory. Vol. 4. No. 1 (Jan 1994). 11-40.

(Assignment.pdf)

[13] Sonmez. Tayfun. Unver. M. Utku. *Course Bidding at Business Schools*. December 21, 2007.

(coursebidding.pdf)

[14] CSBA AEC: California School Boards Association Annual Education Conference and trade show.

E.3 Generalized Assignment Problem

(https://en.wikipedia.org/wiki/Generalized_assignment_problem), where Students are Agents, Events (Courses) are Tasks, and Student Preferences are Costs.

E.4 Random Serial Dictatorship

(https://en.wikipedia.org/wiki/Random_serial_dictatorship) aka Random Priority.

[15] Abdulkadiroglu. Atila. Sonmez. Tayfun. *Random Serial Dictatorship and the Core from Random Endowments in House Allocation Problems*. Econometrica. **66** (3): 689.

[16] Bade. Sophie. *Random Serial Dictatorship: The One and Only*. January 13, 2006. University of London and Max Planck Institut for research on Collective Goods.

(RandomSerialDictatorship-TheOneAndOnly.pdf)

[17] Bichler. M. Merting. S. Uzunoglu. A. *Assigning Course Schedules: About Preference Elicitation, Fairness, and Truthfulness*. Cornell University. 6 Dec 2018.

(Bichler-Cornell-FCFS-FairnessTruthfulness.pdf)

E.5 Fair Random Assignment

(https://en.wikipedia.org/wiki/Fair_random_assignment).
(https://en.wikipedia.org/wiki/Fair_item_assignment). Note that Fair Item Assignment is *not* applicable due to intractability of the combinatorial problem of listing and assessing all available package combinations.

E.6 Rank Maximal Allocation

(https://en.wikipedia.org/wiki/Rank-maximal_allocation)

[18] Irving. R.W. *Greedy matchings*. University of Glasgow. Computing Science Department Research Report. TR-2003-136. April 2003.
(GreedyMatchings-RobertIrving.pdf)

[19] Irving. R.W. Kavitha. T. Mehlhorn. K. Dimitrios. M. Paluch. K. *Rank Maximal Matchings*. University of Glasgow.
(RankMaximalMatchings-journal.pdf)

[20] Irving, R.W. Manlove, D.F. *Finding large stable matchings*. Journal of Experimental Algorithmics. 14.1.2.
(FindingLargeStableMatchings-RobertIrving.pdf)

E.7 Stable Roommates Problem

(https://en.wikipedia.org/wiki/Stable_roommates_problem) Note that Stable Roommates is *not* applicable, since students and Events/classes/courses are distinct “classes” or “categories”.

E.8 National Resident Matching Program

(https://en.wikipedia.org/wiki/National_Resident_Matching_Program). Note that the National Resident Matching Program is *not* applicable since the two categories (students & Events vs. residents and hospitals) in this project don’t desire each other, it’s only a one-way preference: students have preferences for Events.

E.9 Sports Team Selection and Moneyball

[21] Ahmed. Faez. Jindal. Abhilash. Deb. Kalyanmoy. *Cricket Team Selection Using Evolutionary Multi-objective Optimization*. Kanpur Genetic Algorithms Laboratory. Indian Institute of Technology. Kanpur, Kanpur, India. 2011.

(CricketTeams-PerformanceAndBudgetOptimization.pdf)

[22] Anagnostopoulos, A. Cavallo. R. Leonardi. S. Sviridenko. M. *Bidding Strategies for Fantasy-Sports Auctions*. Web and Internet Economics (WINE 2016). Lecture Notes in Computer Science. vol 10123. Springer. Berlin, Heidelberg. Cai. Y. Vetta. A. (eds).

(anagnostopoulos-fantasySports-auctions.pdf)

[23] Dadeloa. S. Turskis. Z. Zavadskas. E. Dadeliene. R. *Multi-criteria assessment and ranking system of sport team formation based on objective-measured values of criteria set*. Expert Systems with Applications. Volume 41, Issue 14. 15 October 2014. Pages 6106-6113.

[24] French. Ryan. *Fantasy Football Player Auctions*. Thesis, Chapman University. 2016.

(Ryan-Chapter1.docx and Ryan-Chapter2.docx)

[25] Hakes. Jahn K. Sauer. Raymond D. *An Economic Evaluation of the Moneyball Hypothesis*. Journal of Economic Perspectives. Vol 20, Number 3, Summer 2006. pages 173-185.

(MoneyballEvalualtion-Hakes-Sauer.pdf)

[26] Lewis. Michael. Moneyball: The Art Of Winning An Unfair Game. W.W. Norton & Company. 2004.

[27] Omkar. S. *Cricket team selection using genetic algorithm*. International Congress on Sport Dynamics. Melbourne, Australia. 2003. pp. 1–9.

(TeamSelectionGeneticAlgorithm.pdf)

E.10 Assignment Problem Application Areas

[28] Hanson. Robin. *The Policy Analysis Market (A Thwarted Experiment in the Use of Prediction Markets for Public Policy)*. in *Innovations Technology Governance Globalization*. pp. 73–88. doi:10.1162/itgg.2007.2.3.73. Retrieved October 29, 2019.

[29] Ledyard. John. Bossaerts. Peter. Fine. Leslie. *Inducing Liquidity In Thin Financial Markets Through Combined-Value Trading Mechanisms*. in *European Economics Review*. 46(9): 1671-1695. October 2002.

(BondConnect-LedyardBossaerts.pdf)

[30] Ledyard. John. Ishikida. Takashi. Olson. Mark. Porter. David. *Experimental Testbedding of a Pollution Trading System: Southern California's RECLAIM Emissions Market*, in *Research in Experimental Economics*. edited by R. Mark Issac. pp. 185-220. JAI Press, 2001.

(RECLAIM-Ledyard-Ishikida.pdf)

[31] Ledyard. John. Olson. Mark. Porter. David. Swanson. Joseph A. Torma. David P. *The First Use of a Combined-Value Auction for Transportation Services*. in *Interfaces*. 32(5):4-12. September - October 2002.

(SearsTrucking-Ledyard-Porter-Olson.pdf)

[32] Ledyard. John. Porter. David. Rangel. Antonio. *Using Computerized Exchange Systems to Solve an Allocation Problem in Project Management*. *Journal of Organizational Computing*. Vol. 4, Number 3. 1994. pp. 271-296.

(CassiniResourceAllocation-Ledyard-Porter.pdf)

[33] Rassenti. S. J. Smith. V. L. Bulfin. R. L. *A combinatorial auction mechanism of airport time slot allocation*. Bell Journal of Economics 13. 402–417. 1982.

[34] Senkul. Pinar. Toroslu. Ismail H. *An architecture for workflow scheduling under resource allocation constraints*. Information Systems. Vol 30, Issue 5. July 2005. pages 399-422.

E.11 Behavioral Economics

[35] Thaler. Richard H. Misbehaving: The Making of Behavioral Economics. WW Norton & Company. May 2015.

F. Crowdsource Tool

The Crowdsource Tool user guide is from the document "Crowd Source State Team Selection.docx".

The contents are included herein:

Crowd Source State Team Selection

The web site should be viewed using Firefox or Chrome browsers. Internet Explorer does not display the page properly (it loads glacially slow and does not work correctly if/when it finished loading).

This year, CVMS is using a web site to assist with the selection of the State Team. Should CVMS students at the Regional competition earn the opportunity to advance to the So Cal State Competition, any student or parent of a student at CVMS is welcome to work to compose a team from CVMS students to be a candidate for the CVMS State Team. A team will be considered a candidate team if the score of that team *exceeds* the score of the current best candidate team, and the person composing that team will be allowed to submit their candidate team for review and validation. If the submitted team meets all of the requirements[169] for a team and the submitted team's score exceeds the current best candidate team, then the submitted team will become the current best candidate team. Any new candidate team submission must now exceed the score of the new best candidate team. At the end of the selection window, the current best candidate team will become the actual State Team.

How does the web site work? Refer to the screen shot of the web site on the following page.

The top rows show the current best solutions, listed from highest to lowest, giving the candidate team's score and who submitted the team. In this example, the teams listed are: Team Einstein @ 302, the actual State Team from last year @ 349.75, and a team selected by the team selection algorithm over this past summer @ 364.75.

Double clicking on one of these teams will cause that team to be loaded into the table below as your current working team.

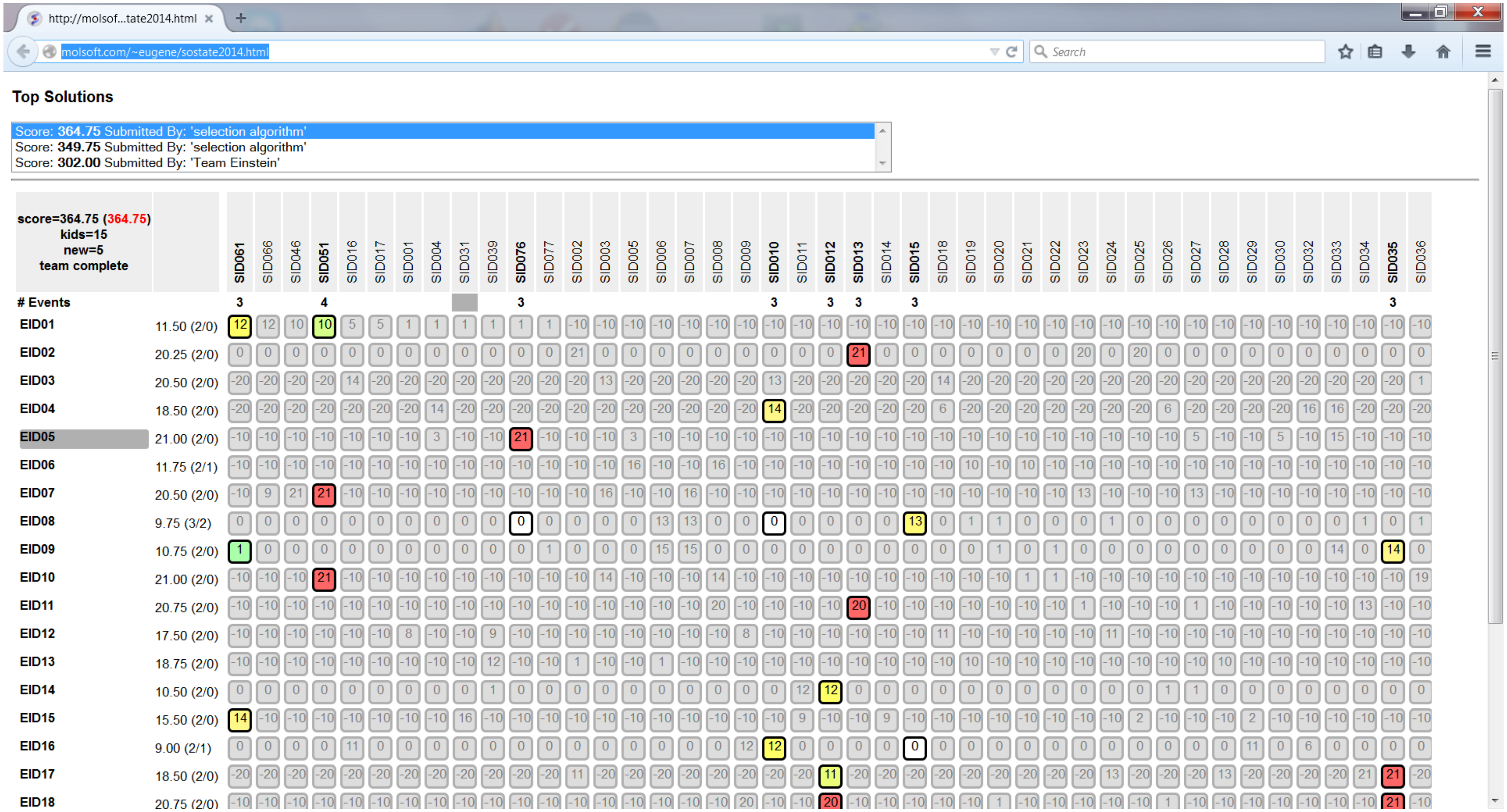
The table is laid out as follows:

The rows are the Events. Each Event is identified by an ID number. The Event ID is in the first column. Double clicking on the Event ID will cause the columns in the table to sort from highest score to lowest score, where the column closest to the Event ID will have the highest score. Where there is a tie score, there is no tie-breaker used in the sort, so any student column with a tied score for the Event that was sorted will be randomly ordered.

---

[169] In addition to student count and Event counts, there are several other considerations that must be taken into account. Some of these could not be incorporated into this web page, including Event day scheduling conflicts between Events. Thus, it is conceivable that a submitted candidate team with a higher score than the current best candidate team may be *rejected* due to other constraints. If this occurs, the reason for the rejection will be reported to the submitter. Note that the State and National competition day Event conflicts are *different* from the Regional competition day Event conflicts.

The columns are the students. Each student is identified by an ID number. The Student IDs are in the top row. The row directly below the student IDs is the total number of Events to which a particular student is assigned. The maximum number of Events a student may have is 4.



score=364.75 (364.75) kids=15 new=5 team complete		SID061	SID066	SID046	SID051	SID016	SID017	SID001	SID004	SID031	SID039	SID076	SID077	SID002	SID003	SID005	SID006	SID007	SID008	SID009	SID010	SID011	SID012	SID013	SID014	SID015	SID018	SID019	SID020	SID021	SID022	SID023	SID024	SID025	SID026	SID027	SID028	SID029	SID030	SID032	SID033	SID034	SID035	SID036
# Events		3			4							3									3		3	3		3																	3	
EID01	11.50 (2/0)	12	12	10	10	5	5	1	1	1	1	1	1	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10
EID02	20.25 (2/0)	0	0	0	0	0	0	0	0	0	0	0	0	21	0	0	0	0	0	0	0	0	0	21	0	0	0	0	0	0	0	20	0	20	0	0	0	0	0	0	0	0	0	0
EID03	20.50 (2/0)	-20	-20	-20	-20	14	-20	-20	-20	-20	-20	-20	-20	-20	13	-20	-20	-20	-20	-20	13	-20	-20	-20	-20	-20	14	-20	-20	-20	-20	-20	-20	-20	-20	-20	-20	-20	-20	-20	-20	-20	-20	1
EID04	18.50 (2/0)	-20	-20	-20	-20	-20	-20	-20	14	-20	-20	-20	-20	-20	-20	-20	-20	-20	-20	-20	14	-20	-20	-20	-20	-20	6	-20	-20	-20	-20	-20	-20	-20	6	-20	-20	-20	-20	16	16	-20	-20	-20
EID05	21.00 (2/0)	-10	-10	-10	-10	-10	-10	-10	3	-10	-10	21	-10	-10	-10	3	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	5	-10	-10	5	-10	15	-10	-10	-10
EID06	11.75 (2/1)	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	16	-10	-10	16	-10	-10	-10	-10	-10	-10	-10	-10	10	-10	10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10
EID07	20.50 (2/0)	-10	9	21	21	-10	-10	-10	-10	-10	-10	-10	-10	-10	16	-10	-10	16	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	13	-10	-10	-10	13	-10	-10	-10	-10	-10	-10	-10	-10
EID08	9.75 (3/2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13	13	0	0	0	0	0	0	0	13	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	1
EID09	10.75 (2/0)	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	15	15	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	14	0	14	0
EID10	21.00 (2/0)	-10	-10	-10	21	-10	-10	-10	-10	-10	-10	-10	-10	-10	14	-10	-10	-10	14	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	1	1	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	19
EID11	20.75 (2/0)	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	20	-10	-10	-10	-10	20	-10	-10	-10	-10	-10	-10	-10	1	-10	-10	-10	1	-10	-10	-10	-10	-10	13	-10	-10
EID12	17.50 (2/0)	-10	-10	-10	-10	-10	-10	8	-10	-10	9	-10	-10	-10	-10	-10	-10	-10	-10	8	-10	-10	-10	-10	-10	-10	11	-10	-10	-10	-10	-10	11	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10
EID13	18.75 (2/0)	-10	-10	-10	-10	-10	-10	-10	-10	-10	12	-10	-10	1	-10	-10	1	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	10	-10	-10	-10	-10	-10	-10	-10	-10	10	-10	-10	-10	-10	-10	-10	-10
EID14	10.50 (2/0)	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	12	12	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
EID15	15.50 (2/0)	14	-10	-10	-10	-10	-10	-10	-10	16	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	9	-10	-10	9	-10	-10	-10	-10	-10	-10	-10	-10	2	-10	-10	-10	2	-10	-10	-10	-10	-10	-10
EID16	9.00 (2/1)	0	0	0	0	11	0	0	0	0	0	0	0	0	0	0	0	0	0	12	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	0	6	0	0	0	0
EID17	18.50 (2/0)	-20	-20	-20	-20	-20	-20	-20	-20	-20	-20	-20	-20	11	-20	-20	-20	-20	-20	-20	-20	-20	11	-20	-20	-20	-20	-20	-20	-20	-20	-20	13	-20	-20	-20	13	-20	-20	-20	-20	21	21	-20
EID18	20.75 (2/0)	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10	20	-10	-10	20	-10	-10	-10	-10	-10	1	-10	-10	-10	-10	-10	1	-10	-10	-10	-10	-10	-10	-10	21	-10

The upper left corner of the table gives status info on the team selection as it stands in the table.

- The score in Black font is the score of the current working team shown in the table.
- The score in the red font is the target score: this is the score that must be beat in order to have a valid candidate team.

- "Kids" gives the total number of students currently on the working team. A maximum of 15 students may be on a team. A team may have fewer than 15 students.
- "New" indicates the number of new Events that students have been assigned. Students have scores from the Events that they participated in this year. Any Event that they did not participate in will have a zero or negative score. If a student is assigned to an Event in which they did not participate in this year, this is considered a new Event and will be tallied in this field. Typically, a higher team score can be achieved by keeping the "New" tally as small as possible.
- "Team Complete" or "Team Incomplete". This status indicates whether or not a team has the correct number of students in all of the Events.

The second column provides scoring and student count information for each Event. The first number is the composite score of the students selected for that Event. The numbers in parentheses show the number of students currently assigned for that Event and then, after the slash character, the number of students with no experience in that Event. A fully assigned team will show zero in the second number for all Events if there are experienced students assigned to the Event. The number will be non-zero if a student with a zero or negative score is assigned to that Event. The sum of the Event scores is the team score that is displayed in the upper left cell of the table.

The main body of the table shows the scores of the various students in their various Events. A higher number means higher performance results in that Event. For this sample screen, the maximum possible number is 21, the lowest possible number is -20. Students who participated in the Event will have numbers ranging from 21 to 1, depending on their team's performance in each Event. The numbers are color coded for easier identification, where red is high, then yellow, then green. White is reserved for 0 and negative scores.

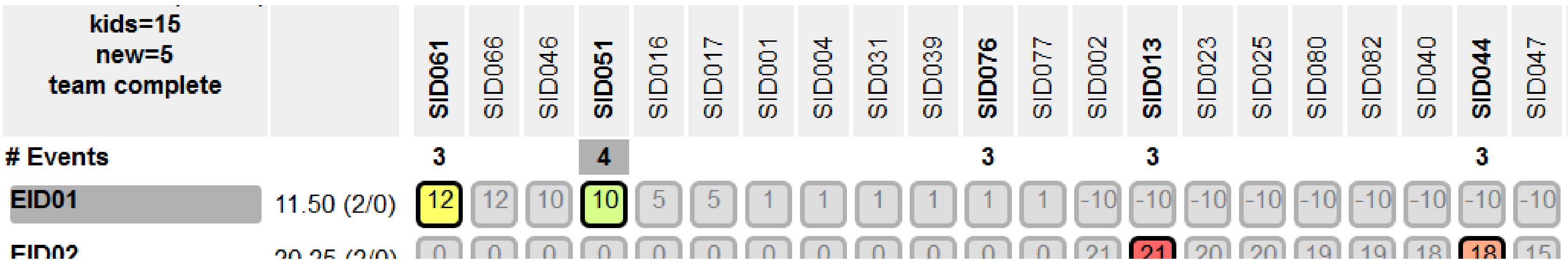
Zero and negative scores. For any Event in which a student did not participate, they are assigned a zero or negative value. For this sample data, build Events are assigned a 0, study Events a -10, and lab Events a -20. This has to do with the competition rules and experience of students in Events. For build Events, only one student need have built the device that is used, the second student is an assistant to the builder[170]. For study Events, the new student must learn the study material, which is a difficult but not insurmountable task, hence a negative score of -10. For lab Events, the new student must not only learn the material, but must also learn the lab techniques involved in the Event, hence a negative score of -20.

How does it work?

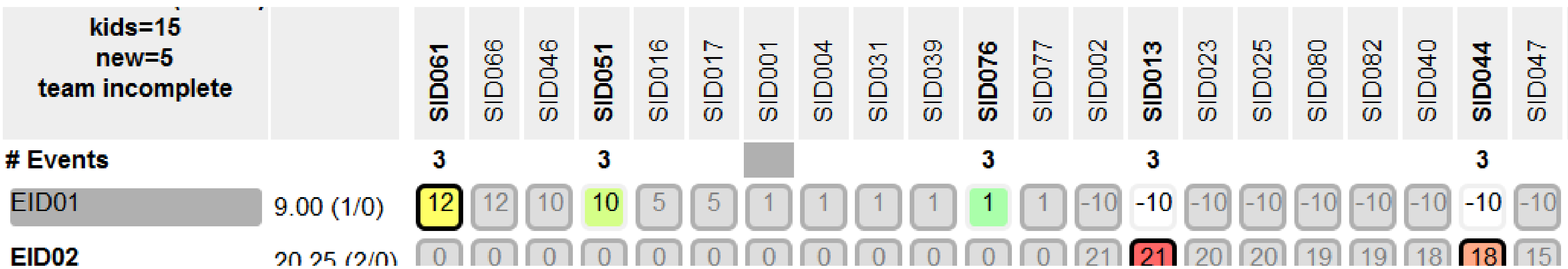
The web page loads with the current best candidate team. The selected students have their IDs displayed in bold (and those not on the team are regular font). The Events to which that student is assigned are displayed by having a border drawn around the Event score associated with that student.

You must first remove a student's Event or a complete student from the team before you may add an Event to another student. To do this, click on one of the black-bordered cells in a student column. For example, click on the Green 10 in the screen shot below.

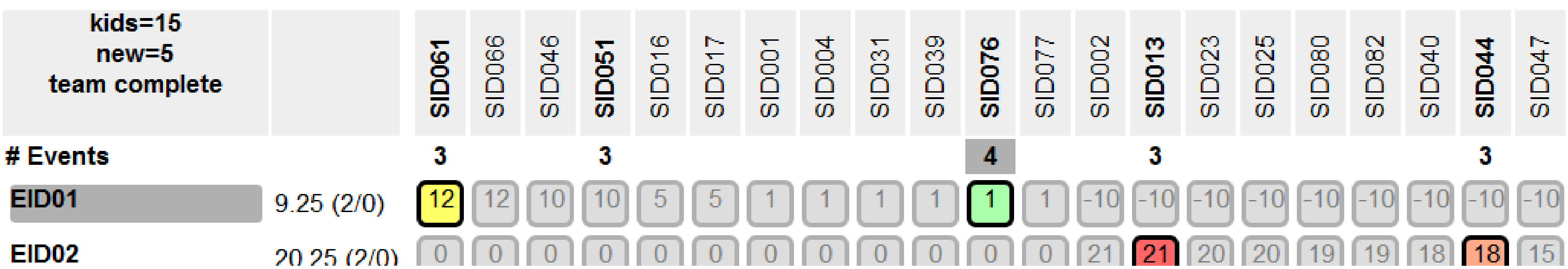
[170] Note that if two new students are assigned to a build Event, they must build the device from scratch, they are ***not*** allowed to use a device built by other students. This means that if two new students are assigned to a build Event, the web site assesses a penalty score of 0 for those students, where in reality this does not reflect the true disadvantage the whole team would face in this situation. This is a consideration that could cause a submitted candidate team to be rejected.



The black border is removed, and all current members of the team are now highlighted in this Event row to indicate they are eligible to be assigned this Event and their score for this Event.



Clicking on this Event under a different student will now assign this Event to that student. In this case the green 1 under student ID 076.



The score for the Event is updated with the new student assignment. In this example, the score for this Event went down, since a student with a score of 10 in this Event was replaced with a student with a score of 1.

To remove a student from the team, click on all of their black-bordered Events. This will then open up a number of other students to be selected for the team. The new student is assigned to the team by clicking on the cells in the new student's column for the Event(s) that are open for assignment.

If a candidate team is arrived at that exceeds the current best candidate team's score, a submit button will appear on the screen and allow you to submit your current candidate working team. You will also be prompted for you to enter your name. This way, you will get credit for submitting a candidate team (and we will know how to get in touch with you if needed).

Please note that there is no mechanism for automatically saving your work and continuing at a different time. If you leave the web site, your work will be lost and **cannot** *be recovered. Period.* So, if you lose your work, please do not ask us to recover it—it is gone into the bit bucket never to return.

Lastly, with features, there is the ability to import team compositions at the bottom of the web page. Here, you may enter a string format with the configuration of a team and have it displayed and scored by the web site. The format of the text string is:

EventID0:StudentID1,StudentID2 EventID1:StudentID1,StudentID2 (etc)

If you want to save your work, you can manually record your Event and student IDs into a text file and then copy and paste it back in to pick up where you left off.

For those of you with the time and inclination, you can write your own team selection algorithm, export your results into the above format and import it into this web page and see how your algorithm does.

Interestingly, the human mind is really bad at building a highly scored team given this type of problem when they try to do it from scratch. For example, the team with a score of 349 was made entirely by hand, but took 3 people working over 28 hours each to arrive at the solution. The computer algorithm arrived at the team with a score of 364 in about 5 hours on over 500 CPUs.

More interestingly, the human mind is amazingly adept at taking a good solution and making it *better*. One parent then took the computer selected team and improved on it by about 5% in under an hour.

So, here is the challenge: we know there is at least one team that is 5% better than the current best candidate team. In fact, it is very likely that there are at least two teams with better scores than the current best candidate team, where the second team is likely at least 10% better scoring than the current best candidate team.
Let's see if anyone can find at least one of the better teams.

As a final note, remember that this web site was built by volunteers on their personal time. While the web site appears to do all that it needs to do for this application of team selection, there may be bugs or other issues with the software. For example, we have never used it with more than 4 or 5 people using it at the same time, so there may be problems that arise when 100+ people try to hit it at the same time.

While we are happy to receive any feedback on the site, especially any bug reports, please be aware that we may not have the time to make fixes to any reported bugs. It is very unlikely that we would have the time to add features that some people may suggest—even if the suggester thinks it is ‘critical’ that a particular feature be included.

Have fun, and we hope at least a couple people are able to find the better team(s).