Acta Astronautica 83 (2013) 145–174

Contents lists available at SciVerse ScienceDirect

Acta Astronautica

journal homepage: www.elsevier.com/locate/actaastro

Collaborative gaming and competition for CS-STEM education using SPHERES Zero Robotics ☆

Sreeja Nag *, Jacob G. Katz, Alvar Saenz-Otero

Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

ARTICLE INFO

Article history:
Received 18 March 2012
Received in revised form
27 July 2012
Accepted 15 September 2012
Available online 20 November 2012

Keywords:
Collaborative gaming
Educational Robotics
Programming competitions
Social science research

ABSTRACT

There is widespread investment of resources in the fields of Computer Science, Science, Technology, Engineering, Mathematics (CS-STEM) education to improve STEM interests and skills. This paper addresses the goal of revolutionizing student education using collaborative gaming and competition, both in virtual simulation environments and on real hardware in space. The concept is demonstrated using the SPHERES Zero Robotics (ZR) Program which is a robotics programming competition. The robots are miniature satellites called SPHERES—an experimental test bed developed by the MIT SSL on the International Space Station (ISS) to test navigation, formation flight and control algorithms in microgravity. The participants compete to win a technically challenging game by programming their strategies into the SPHERES satellites, completely from a web browser. The programs are demonstrated in simulation, on ground hardware and then in a final competition when an astronaut runs the student software aboard the ISS. ZR had a pilot event in 2009 with 10 High School (HS) students, a nationwide pilot tournament in 2010 with over 200 HS students from 19 US states, a summer tournament in 2010 with ~150 middle school students and an open-registration tournament in 2011 with over 1000 HS students from USA and Europe. The influence of collaboration was investigated by (1) building new web infrastructure and an Integrated Development Environment where intensive inter-participant collaboration is possible, (2) designing and programming a game to solve a relevant formation flight problem, collaborative in nature—and (3) structuring a tournament such that inter-team collaboration is mandated. This paper introduces the ZR web tools, assesses the educational value delivered by the program using space and games and evaluates the utility of collaborative gaming within this framework. There were three types of collaborations as variables—within matches (to achieve game objectives), inter-team alliances and unstructured communication on online forums. Simulation competition scores, website usage statistics and post-competition surveys are used to evaluate educational impact and the effect of collaboration.

1. Introduction

Space has often been considered a hobby of the intellectual elite. It is to dispense this myth, and utilize space as a perfect and accessible laboratory environment where the science equations in textbooks come alive, that outreach and education programs that engage public are required. Literature and current market trends (discussed in the forthcoming sections) have amply pointed out how games bring out the best in people in terms of learning and productivity. Gaming, in the obvious sense, is the act of playing a game. Games have now transcended the bounds of virtual reality and entered our lives and the lives of

☆ This paper was presented during the 62nd IAC in Cape Town.

* Corresponding author. Tel.: +1 61 7710 1845.
E-mail address: sreeja_n@mit.edu (S. Nag).

0094-5765/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.actaastro.2012.09.006

others we know. Interactions in the gaming world translate into interactions in the real world. To play the game better, thus, needs collaborations in both worlds.

CS-STEM – an acronym for Computer Science (CS), Science, Technology, Engineering and Mathematics – education refers to efforts invested in bringing students and young professionals, the next generation workforce, up to speed in the fields of CS-STEM and therefore be prepared to address the grand challenges of the 21st century. A recent editorial in the Science Magazine [1] defined STEM Education as, "*For most, it means only science and mathematics, even though the products of technology and engineering have so greatly influenced everyday life. A true STEM education should increase students' understanding of how things work and improve their use of technologies. STEM education should also introduce more engineering during precollege education. Engineering is directly involved in problem solving and innovation, two themes with high priorities on every nation's agenda. Given its economic importance to society, students should learn about engineering and develop some of the skills and abilities associated with the design process.*" Given the current generation's dependence on digital and media technologies, a nation's economy depends upon its people's ability to contribute computationally to its challenges. Computer science has moved up the ranks rapidly and found its spot as an important part of STEM education.

This paper introduces collaborative games as the bridge between space-based engineering and CS-STEM education.

1.1. CS-STEM and space education

The idea that a revolutionary, hands-on method of education is required to create and maintain students' interest in STEM[1] has been floating about since decades. Children learn by doing and thinking about what they do, an idea supported by Minsky [2] who believes that good STEM education is not only that which teaches students to use and learn about new technology, but also gives them the tools to modify technology to suit their own needs.

The United States, in spite of being the largest spender on education in the world has among the lowest Science and Math test scores in the developed world. The 2007 TIMSS[2] scores showed that 15% of the US fourth-graders and only 10% of U.S. eighth-graders scored at or above the advanced international benchmark in science [3]. An infographic prepared by the University of Southern California using data published by the OECD [4], CIA [5] and UN from 2003 through 2009 compares the education spending in 12 countries with their corresponding literacy rate, school life expectancy and math and science test scores (standardized). The students surveyed for the figure for test scores were 15 years old. It shows that although the USA is the highest spender at the highest per school child expenditures of $7743, it is #3 from the top in literacy rate, #5 in school life expectancy and #7 and #8 in science and math scores respectively. While hiking up U.S. performance in math and science is important, computer science skills also require a special mention here. "21st Century skills", the new buzzword in education, refers to a growing global movement to redefine the goals of education, to transform how learning is practiced each day, and to expand the range of measures in student achievement, all in order to meet the new demands of the 21st Century [6]. Literature shows that incorporating critical thinking, problem solving and communication into the teaching of core academic subjects is indispensible to 21st century learning. Moreover, the three core skills [7] required are:

1. Life and career skills (flexibility, adaptability, initiative, self direction, communication, social and cross-cultural interaction, productivity and accountability, and leadership and responsibility).
2. Learning and innovation skills (critical thinking and problem solving, communication and collaboration, and creativity and innovation applied to imagination and invention).
3. Information media, digital media and technology skills.

Note that 21st century skills call upon not only pure math and science but the ability to use and manipulate computer technology. To enforce the point further, while the 3 R's of "reading, 'riting, and 'rithmetic" were deemed essentials of mandatory public schooling in the 19th century, 21st century literacy is *defined* by 4 R's [8]: Reading, 'riting, 'rithmetic and 'rithms, the fourth R being algorithms or basic computational skills. Therefore, any sort of STEM education effort should certainly involve computer science and media interaction (i.e., CS-STEM) as well as promote communication, leadership, critical thinking, imaginative problem solving as soft skills.

Space exploration and technology has always been a region of fantasy to everyone, especially children, and space-related activities are therefore excellent motivators to learning and fostering interest in STEM. Not surprisingly, education of the next generation workforce has always been one of NASA's mission goals. Two of six goals released as part of NASA's 2011 Strategic Plan have direct relevance to STEM and education [9]. For instance, Goal 6 states: "*Share NASA with the public, educators, and students to provide opportunities to participate in our mission, foster innovation and contribute to a strong National economy.*" It directly calls upon the agency to create opportunities for broad outreach and student involvement in projects. Goal 3: "*Advance aeronautics research for societal benefit*" indirectly refers to educational advancement too, since a society's future depends on the education of its citizens and their ability to use their education to contribute to the economy.

[1] The terms STEM and CS-STEM are used interchangeably. The intent is always to mean CS-STEM since computing is now considered an indispensible 21st century skills (will be explained in this section).

[2] The Trends in international Mathematics and science study (TiMss) is an international assessment and research project designed to measure trends in mathematics and science achievement at the fourth- and eighth-grade levels as well as school and teacher practices related to instruction. Since 1995, TIMSS has been administered every 4 years.

Since the 1980s, NASA has played a very beneficial role in directed space education outreach in the United States, inspiring students and teachers across the nation. Two of NASA's largest educational programs: the NASA Explorer Schools (NES) and NASA Spaceward Bound programs are examples of outreach (in the past and currently ongoing) to promote student interest in science, technology, engineering, math, and geography (STEM-G) careers [10]. The ISS since its starting stages has been extensively used to conduct research by universities, and extra effort is being invested in getting students involved with the onboard activities. The NASA "International Space Station Education Concept Development Report [11] states: *"Utilizing the International Space Station National Laboratory for education is an effort initiated in response to the 2005 NASA Authorization Act, which designated the U.S. segment of the ISS as a national laboratory"*. The report reveals a framework where goals are laid out in a pyramid structure: inspire a large number of students, engage a set of them, and educate a sub-set of these. However, a truly revolutionary education program should and would inspire a large number of students, allowing many to learn by directly *engaging* them. Since it began operations, the ISS has accommodated a number of education experiments. *Engagement* is a critical first step for education and while multiple programs have reached a substantial number of students via demonstrations and videoconferences with astronauts, these have traditionally not allowed students to become *engaged* in actual research activities, but represented more or less a one-way flow of information. Bob Rogers, founder and Chairman of BRC Imagination Arts and winner of the NASA Public Service Medal, when developing NASA's master plan for the exploration of Mars as part of the Mars Exploration Program Analysis Group, presented five strategies for public and student engagement [12]. The presentation, summarized by Mark Craig, makes three important points for effective *engagement*:

Effective and massive public engagement has important benefits beyond increased support. It enhances work force retention, morale and recruiting because "It's nice to be a part of something famous". It enhances "spin control" of unplanned events because it establishes a compelling context. The most profound benefit is that it builds a "psychological highway to space". If done well, public engagement builds the exploration and opening of the space frontier into the Nation's DNA. Engagement is best achieved to the broadest audience through the use of a 'story'. As people are engaged by a story, goals in the story need only be important to the protagonists (us). Said in reverse, if people are not engaged by a story, explaining why our goals should be important to them will never be enough. "Story" is an effective mechanism for dealing with potential showstoppers such as loss of interest after major accomplishments (Apollo 12 syndrome). It is also key in sharing the experience of space exploration because it takes people with us emotionally, beyond just visual and tactile experiences.

Important components in making great education possible are international collaboration in development of interest in space, and providing easily accessible information and development of programs that will motivate the next generation workforce. Space Exploration educators across the globe are confronting challenges and embracing opportunities to educate and prepare students for an increasingly interconnected world. Collaboration is in the interest of the US as well. A recent National Research Council (NRC) Space Studies Board report [13] acknowledges that "*US problems requiring best efforts to understand and resolve are global in nature and must be addressed through mutual worldwide action*". The report notes that educating "*a capable workforce for the 21*st *century is a key strategic objective for the US space program*". It further recommends that the International Space Station (ISS) be utilized fully for education and research, echoing a similar educational recommendation in the Augustine Commission Report [14].

1.2. Collaborative gaming and competition

Games have been around as long as human history has been documented. They allow us to build worlds that specifically tap into our evolutionary senses. Stuart Brown [15] observed animal play in the wild, where he first conceived of *play as an evolved behavior* important for the well-being and survival of animals, especially those of higher intelligence. Play, he concluded, has been known to pique human curiosity (exploration play), cause community collaboration (social filling play), charge better performances (adrenaline pumping play) and bring out the creative best in people (imaginative play). Jane McGonigal from 42 Entertainment that produced the record-breaking *'I love Bees'* has researched the reasons for games bringing out the best in people [16]. The positive outcomes of games, she suggests, are blissful productivity, urgent optimism, working in a collaborative environment and toward something agreed upon as an 'epic win'. Furthermore, the common theme among all the gaming blockbusters of today is the fact that they all break into reality [17]: FarmVille lets Facebook users play with their real friends, Guitar Hero lets music lovers play the game while playing music real-time on a real instrument, Nintendo Wii or the Microsoft Kinect use a real console to translate real actions into a video game. The internet, being the best platform for broadcast as well as conversation, has been the critical facilitator of games entering real lives of communities of people worldwide. The introduction of reality in games – picked up and virally spread by alternate reality games– has made the reasons to play them stronger and shown strong correlation between behavior in games to rational, economic behavior in real life [18]. Games are great tools to pique human productivity and reward the brain [19] because they provide easy-to-monitor bars of progress (e.g., An evolving Avatar), multiple short and long term aims, an easy link of consequences to actions, elements of uncertainty to keep the user's interest, windows of enhanced attention as users race for a predefined goal and a crowd of players to play with or against.

In the context of this paper, gaming is defined as the act of playing a game using an online interface or inside a virtual world. (MMORPGs). MMORPG is a genre of role-playing video games in which a very large number of players interact with one another within a virtual game world. Revenue for the gaming industry is generated largely through subscriptions and sometimes through advertising. In 2008, the consumer subscription spending on subscription MMORGGs in North America and Europe was over $1.4 billion [20].

Competitions based on the concept of games can organize individuals to work toward a common objective with the incentive of a monetary or non-monetary reward. Individuals with a diversity of skills can participate in the task, with participants picking up and contributing in tasks they are best at. Collaboration allows individuals to work together to achieve larger goals. However, meaningful development through competitions requires a careful balance of competition and collaboration to achieve its goals. One of the important tenets of this paper is that competition and collaboration are not mutually exclusive. While big competitions 'challenge' the public with a difficult objective, a series of smaller challenges can be used to engage multiple participants if the challenge structure includes collaboration. Collaboration among the participants allows for the accomplishment of larger tasks by multiple people, and for the performance of each participant to be improved by learning from others. There are a number of ways to bring collaboration into a competitive model, while retaining the benefits of competition. MMORPGs always have the common feature of social interaction. The games are designed such that some degree of team work is required in order to achieve game objectives. Strategies are decided upon by communication via typed conversation and due to the large online forum available, players often find like-minded players to collaborate with. While some individuals may be outcasts in the real world, they can become whomever they want in these virtual worlds, and can find other players with similar interests and personalities. In one survey, 39.4% of males and 53.3% of females felt that their MMORPG companions were comparable to or even better than their real world friends [21].

Breaking all these virtual, collaborative games into reality, while keeping the excitement and story mentioned above, is the concept of Alternate Reality Games (ARGs) [22] e.g., *I love Bees* from 2004 which had over 600,000 players. ARGSs have an "*interactive narrative that uses the real world as a platform and uses transmedia to deliver a story*" that may be altered by participants' ideas or actions in the virtual or real world. Players interact directly with characters in the game, solve plot-based challenges and puzzles, and collaborate as a community to analyze the story and coordinate real-life and online activities. ARGs generally use multimedia, such as telephones, email and mail but rely on the Internet as the central binding medium. The stereotype of a gamer as a lone and asocial individual has been disproven [23]. On personality tests, gamers have proven to be more extroverted, open, and conscientious than non-game players [24]. Moreover gamers prefer to play with people they already know turning the game into a social experience and may even make, confirm and maintain friendships and relationships through gaming [25]. In summary, gaming has become a collaborative phenomenon to achieve the required game objectives and is far more than adversarial competition. Such games provide tremendous potential to tap into the several million strong gaming community worldwide to help solve puzzles when judiciously articulated in the language of the game (objectives, incentives, rules etc.).

2. SPHERES Zero Robotics program

The paper presents the NASA and DARPA supported SPHERES Zero Robotics (ZR) as a revolutionary program that achieves the goals of CS-STEM education by calling upon students to play games with real satellites in space [26]. ZR is an international, robotics programming competition where the robots are SPHERES satellites inside the International Space Station. Tournaments are free of charge and all that is required is a team of students, a mentor and access to a couple of computers with internet connectivity. Students program the robots to play the challenging games, all from a web browser. In the final competition, an astronaut runs the games on the satellites in microgravity and interacts with students live from the ISS (Fig. 9). By leveraging the excitement of the virtual gaming world and providing the reality of astronauts, ISS satellite control and a final showdown event, ZR successfully inspires crowds of students, the way only space can.

The overall goal of the Zero Robotics tournaments is to crowdsource cluster flight algorithms (specifically for SPHERES but that can be generalized to small satellites) and promote STEM education [27,26]. As depicted in Fig. 1, the students who participate in the tournaments are the input into the Zero Robotics 'system' and the output are STEM education and satellite software or algorithms. The outer 'blue box' system therefore has a dual impact of algorithm development *and* education. The system includes a game which is available through the ZR Web Infrastructure (red circle) [28], which in turn is comprised of a website, tutorials, online community forums, team management tools, tournament management and participation tools and programming environment where students can create, save, edit, share, simulate and practice as well as submit computer code for competitions. The feedback of the students, as they participate in the tournaments, serves to improve the web infrastructure (red circle) in Fig. 1. This paper concentrates on only the STEM education goals and benefits of

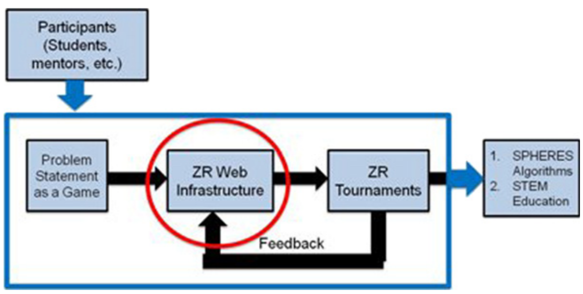

Fig. 1. Zero Robotics system diagram. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

the program, however previous literature may be referred to for deeper insight into the overall program impact [26,28].

2.1. SPHERES

The SPHERES program began in 1999 as part of an MIT Aero/Astro undergraduate class. Prototypes were built by the student class in 2000, flight satellites were delivered in 2003, and launched to the ISS occurred in 2006 [29]. SPHERES became one of the first educational programs that launched student-designed hardware to the ISS. SPHERES consists of a set of tools and hardware developed for use aboard the ISS and in ground-based tests: three nanosatellites, a custom metrology system (based on infrared and ultrasound time-of-flight measurements), communications hardware, consumables (tanks and batteries), and an astronaut interface. They operate aboard the ISS under the supervision of a crew member (Fig. 9, Fig. 2).

The ground-based setup consists of a set of hardware analogous to what is in the Station: three nanosatellites, a metrology system with the same geometry as that on the ISS, a research oriented GUI, and replenishable consumables. Due to gravity the ground-based testbed is implemented on a flat floor, allowing exercising three out of six degrees of freedom. The SPHERES satellites implement all the features of a standard thruster-based satellite bus. The satellites have fully functional propulsion, guidance, communications, and power sub-systems. These enable the satellites to maneuver in six degrees of freedom (6-DOF), communicate with each other and with the laptop control station, and identify their position with respect to each other and to the reference frame. The laptop control station (an ISS supplied standard laptop) is used to collect and store data and to upload new algorithms. SPHERES uploads new algorithms (ahead of time) and downloads data (after the session) using the ISS communications system. Fig. 3 shows a picture of a SPHERES satellite and identifies its main components. Physical properties of the satellites are listed in Table 1. There are two communication channels for data transmission: the SPHERES-to-Laptop (STL) channel to transmit data and telemetry to the laptop station and the SPHERES-to-SPHERES (STS) channel used for inter-satellite communication, enabling cooperative and coordinated maneuvering between satellites during tests. The amount and frequency of data transmission possible with the SPHERES hardware was a limiting constraint in the development of algorithms for the SPHERES and for their usage as robots in a game.

Fig. 2. Astronaut and MIT alum Gregory Chamitoff operates 3 SPHERES aboard the ISS.

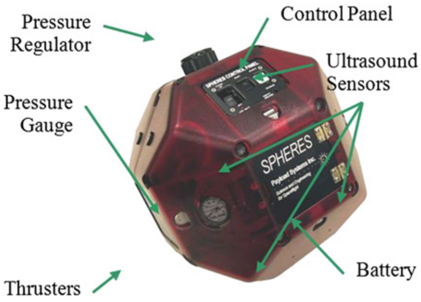

Fig. 3. A SPHERES satellite.

Table 1
SPHERES physical properties.

Diameter	0.22 m
Mass (w/tank & batteries)	4.3 kg
Max linear acceleration	0.17 m/s^2
Max angular acceleration	3.5 rad/s^2
Power consumption	13 W
Battery lifetime (replaceable)	2 h

SPHERES was designed to be a permanent *facility* aboard the ISS, not just a single experiment, by following a set of design principles learned from previous MIT SSL experience [29]. To provide the ability to involve multiple scientists in a simple manner, a SPHERES Guest Scientist Program was created [30]. This program consists of a test development framework, a robust and flexible interface to the SPHERES flight software, a portable high-fidelity simulation, two laboratory test beds and data analysis utilities, and supports the efforts of geographically distributed researchers in the development of algorithms. SPHERES software consists of an embedded system (SPHERESCore) and additional user-selectable library function. SPHERESCore is responsible for handling interrupts and interfacing with the hardware [31]. The library functions such as math utilities, etc., provide guest scientists with the ability to use pre-defined utility functions to expedite programming and testing. The coding language used on the hardware is C, while code for the simulation is in MATLAB. The Zero-Robotics program expands the Guest Scientist Program with a simplified interface and a high-fidelity back-end online simulation so that students at many different grade and skill levels can program the satellites.

2.2. Zero Robotics

Zero Robotics (ZR) is the umbrella program under which multiple *tournaments* are held. A tournament is a series of *competitions* which cater to the same group of participants (e.g., high school students or middle school students) and require one application to be submitted to participate in the tournament. Until 2011, all applications had to be from *teams* not individuals. A *competition* is a bracketed set of matches among the participants (e.g., round robin, double elimination) at the end of which a ranked list can be declared. The participant programs play one game per competition and games may be repeated over multiple competitions. Participants write programs online to play the pre-defined game and submit their program for the purpose of an automated competition. A *match* is a head-to-head run between two SPHERES satellites, in simulation or hardware, controlled autonomously by programs written participants. Typically, opponent players control 2 SPHERES and are each given an automatic score at the end of the match. Section 3 will describe the history of the program, nature of ZR games, the new web interface available to participants to participate on the tournaments and the structure of the tournaments. The Zero Robotics (ZR) competitions draw significant inspiration from FIRST Robotics [32] and share common goals, including building lifelong skills and interest in science, technology, engineering, and math through project-based learning.

In fall 2009, the SSL conducted a pilot program of the Zero Robotics competition with two schools/10 students from northern Idaho [33]. In 2010, Zero Robotics was a component of NASA's Summer of Innovation, a nationwide program targeted at encouraging STEM education for middle school students. During this competition, 10 teams and over 150 students from schools in the Boston area worked for five weeks to program the SPHERES to compete in an obstacle course race. In the fall of 2010, Zero Robotics conducted a nationwide pilot tournament for high school students named the Zero Robotics SPHERES Challenge 2010. Over 200 students from 19 US states participated as part of 24 teams. The objective of the game was to complete the assembly of a fictitious solar power station by maneuvering a satellite to dock with a floating solar panel and then bring it back to the station to finish the mission before the opponent does. In the 2010 tournament, the two SPHERES satellites in each match, controlled by opposing participants, engaged in direct head-to-head competition. Participating teams competed as individual teams throughout the entire tournament, and there were no extensive community forums where they could exchange knowledge or converse with each other. Thus, the 2010 tournament emphasized "pure" competition. An external forum plug-in was provided on the website, but due to the inherent competitive nature of the game, it was not very widely used.

The 2011 tournament was designed to explore crowdsourcing objectives i.e., solve a hard cluster flight problem, as well as STEM objectives, i.e., educate students and outreach. Additionally to investigate the effects of collaborative competition, three different types of collaboration mechanisms were introduced within its tournament structure and games, with the intent of improving the educational experience of participating teams and learning to design future tournaments better, with an appropriate and beneficial mix of collaboration and competition (Table 2). First, the game was designed such that teams that programmed their SPHERES would be encouraged to collaborate during the match to achieve game objectives (i.e., crowdsourcer objectives) and would gain more points than those that did not. Since collaboration was meant to be rewarded more than winning, the competition structure was that of a round robin where the team with the maximum cumulative points won the competition, not the one with the maximum number of wins. Second, halfway through the tournament, there was a mandatory requirement that selected teams had to form alliances of 3 teams each and submit integrated projects per alliance for all competitions after that. Third, the 2011 tournament had extensive community forums where teams could exchange ideas, educate each other, challenge each other to informal games and share projects to work on collaboratively.

3. Components of SPHERES Zero Robotics

The Zero Robotics program has three major components as enumerated within the outer, blue box of Fig. 1: the ZR Web Infrastructure, the problem statement that the students solve in the form of a ZR game and the ZR Tournaments to organize the competitions and select the winner. The components are described below in detail.

3.1. ZR web infrastructure

To allow crowds of students to use the SPHERES high-fidelity simulator, write spaceflight-capable programs and interact/collaborate with each other, an online environment was required—the 'red circle' in Fig. 1. The web interface comprised of a programming interface and several other tools such that participants of the ZR Tournaments could program the SPHERES satellites, submit their programs for competitions and for teams to interact with each other (to achieve the collaboration and competition objective). It also allowed the organizers of the program, the administrators, to conduct tournaments and competitions with thousands of users and hundreds and thousands of simulations entirely online. In this paper, the term 'Web Interface' has been used to refer to the Frond End – the part that the users and administrators interact with – while the term 'Web Infrastructure' has been used to refer to the entire software infrastructure as shown in Fig. 4 along with the website, community forums, tutorials, team and project and tournament management tools.

3.1.1. Programming interface

Typically, programming the SPHERES satellites requires users to have access to the Texas Instrument compilers for the SPHERES processor and familiarity with the Guest Scientist Program. None of this is possible for a tournament meant for high school and middle school students. Instead, a web-based interface was developed to program the satellites which makes use of the same SPHERES high-fidelity simulation that is used to develop flight software.

Users can program the SPHERES using a web-based GUI, which provides a simplified interface to the Guest Scientist

Table 2
Comparison of Zero Robotics competitions in 2010 and 2011 to highlight the introduction of collaborative competition mechanisms.

2010	2011
Purely for STEM Education	For crowdsourcing and STEM
In game competition = > Elimination	In game collaboration = > Round Robin
Individual teams	Individual teams + Alliances
No community forum	Extensive community forums

API functions and enforces constraints that guarantee compatibility with the SPHERES compilers. Students have access to a text-based editor as well as a graphical editor, for those with little or no prior programming experience. A distributed computation engine, hosted on Amazon EC2 virtual machines, compiles the user code with the core SPHERES software, and performs a full simulation of the program. An Adobe Flash-based front-end visualization creates an animated representation of the results. The code programmed by the students via the web interface can be executed in the hardware. The flow of information in the ZR software infrastructure is shown in Fig. 4. The user code is transmitted to the web application, which launches a simulation instance on the 'Farm', which on completion returns the results to the web app and finally the browser, then rendered in the form of an animation as shown in Fig. 5. The ZR 'Farm' is the back-end engine to handle and implement compilation and simulation requests from the web app. The ZR projects are compiled/simulated in conjunction with the SPHERES embedded system (SPHERESCore) code and the ZR game code.

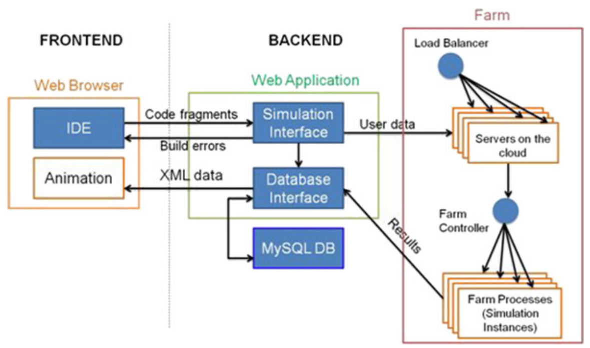

Fig. 4. ZR Web Infrastructure ('red circle' in Fig. 1). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Users write their programs to control SPHERES inside the main function called *'ZRUser()'* available as a template in each project. Users are not allowed to change its signature. *ZRUser()* is called at every iteration of the satellite control cycle (once per second). Users may also declare and define additional procedures, which are all called inside this main loop. The inputs to *ZRUser()*, available to be used by the users, are the SPHERES state (position, velocity, attitude and attitude rates) and the time since the game began. These inputs are obtained from the 'game code', which in turn gets it from the SPHERES embedded system code (explained later in Section 3.2, Fig. 7). For running simulations, the code within *ZRUser()* is inserted into a pre-defined template and simulated by the SPHERES Simulator along with 'game code' and embedded system code. Fresh high school students take less than 3 weeks to learn how to use the IDE and write a fully capable program to play a ZR game.

Graphical Editor: The ZR graphical editor, as shown in Fig. 6, allows users with little or no C experience to write code using drag-and-drop programming. It is currently possible to see and generate C-code from the diagram view so that users can initiate their code with diagrams but can move on to more complicated code using the C editor.

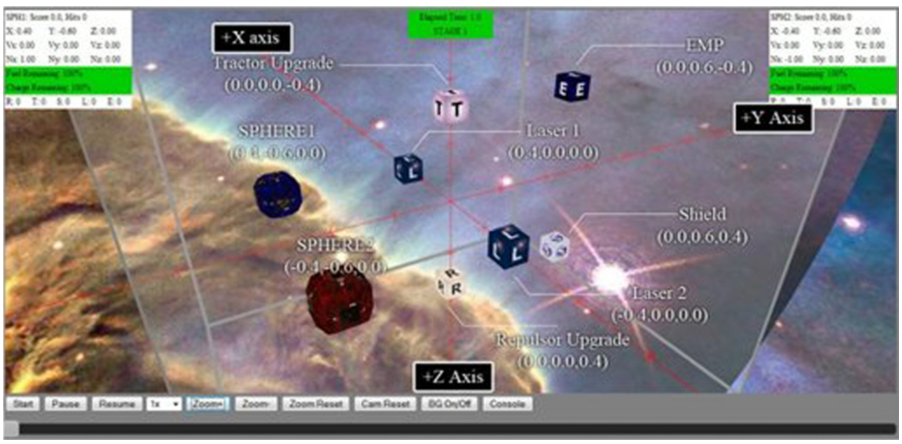

Fig. 5. Example of a ZR Animation.

The graphical editor uses standard procedural language constructs such as if/then/else calls, variable assignments, array iterators, range iterators, case-statements, etc. The Zero Robotics API procedures and functions as well as game specific API functions are integrated into the drag-drop programming icons. Furthermore, user-defined procedures/functions and variables are supported. The graphical editor is written in JavaScript and is derived from the Waterbear JavaScript editor (http://waterbearlang.com). The implementation uses a Model-View-Controller paradigm where the block diagram and "C" views are different renderings of the same underlying model. From past ZR experience (summer program of 2010), middle school students have typically taken less than 10 days to learn to use the graphical editor and submit a program.

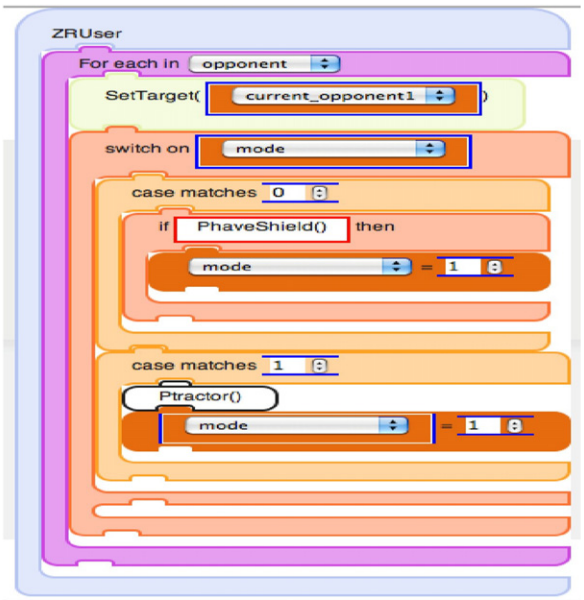

Fig. 6. Example of code in the Graphical Editor.

3.1.2. Team and project management tools

Teams are organized into two types of members: team leads and team members. Users are required to create an account on the ZR website before submitting an application to a tournament. A tournament application for high schools typically comprises of entering school, potential team and mentor information. We also sought a commitment that the students have internet access and have found at least professional individual, affiliated with the school and capable of teaching CS-STEM, who will serve as their 'team mentor'. On acceptance, the user who submitted the application is designated as a team lead of a newly created online team (unique ZR ID assigned) or a previously formed one. A team lead can then invite other users to join the team and assign more team leads. Users have access to a C text editor, a project management tool and a simulation management tool, where the user may replay his past simulations and animations [26]. The project management tool also allows users to navigate and edit projects that have been shared within his team. All users who share a common project have access to the 'project instant messaging' tool so they can chat with each other online while editing their shared projects. No chat logs were saved to protect user privacy.

The ZR simulation allows users to tweak different game parameters and choose simulation settings [26] so that they can test different parts of their code independently. They can simulate an individual project, race against another member of their team or race against standard players (pre-coded projects to simulate against) provided by MIT. The simulation also allows students to control the speed of the game to show the motion in real time, or up to 10 times faster. In a formal competition, these settings are fixed by MIT, and the purpose of the simulation is to provide ample opportunities to test different versions of their strategies and finalize a robust submission. Users may simulate individual projects on the IDE itself, and therefore iterate to improve their projects.

All through the tournaments, teams are given the opportunity to challenge other teams for informal scrimmages. The website provides the ability to select a user project and invite other teams to race their projects against the selected one—called a 'challenge'. Teams can

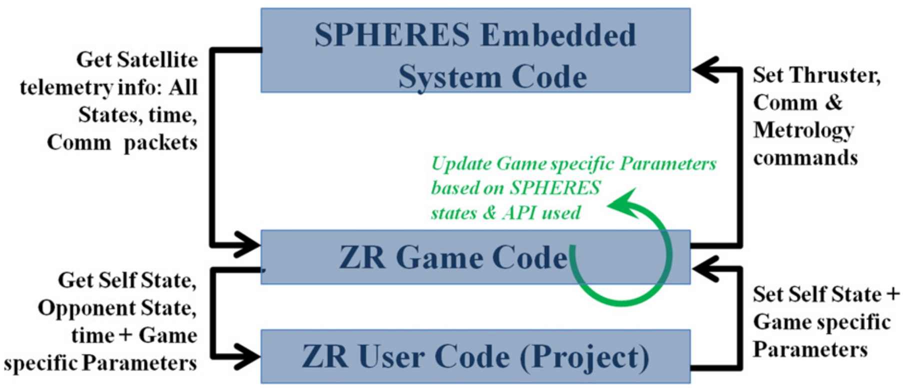

Fig. 7. Block diagram of the flow of information between the three levels of code that make up the spaceflight software that operate *each* SPHERES satellite. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

accept or reject challenges using the provided UI and view the results, animations and leader boards for each challenge that they participated in. The web infrastructure provides for the running of an automated simulation when the challenge has been accepted by a team, and makes the results available on the website. A simple interface is available to teams for submitting a project as an entry into a formal competition—any team lead may select an existing team project and submit it for a competition [26].

3.1.3. Tournament management tools

Administrators need tournament management tools to manage competitions and tournaments on the web interface. They can create an application form for an upcoming tournament, be notified when a user submits and application, review and accept an application—after which the user is automatically emailed a URL which they can use to create their new team or use an existing team for the new tournament. Each tournament may have multiple competitions; each competition can be associated with only one game (See the introduction of Section 4 for details). They may upload game code for any number of games on the web interface, to be associated with specific competitions later or to be simply available on the IDE for users to practice programming. Administrators can create competitions for any tournament, edit its description, set a game to play and set deadlines for the competitions. Users who do not submit their projects by the deadline are disqualified from the competition. To simulate multiple projects, there is a batch simulation tool available to administrators. This tool is very useful for running simulation competitions by simply selecting the user projects to be simulated, the game they intend to play and the game specific parameters. The tool automatically runs thousands of simulations and outputs the results, which the administrator can then make available on the website for users to review and learn from. Administrators may also moderate community forums, make announcements or any changes to the website.

3.2. ZR games

As mentioned before, each tournament unveils one or more software games which the participants are expected to play by programming their robots, the SPHERES satellites, to achieve the game objectives within a predefined period of time. The game objectives are defined and programmed by the game developers at MIT, different for each tournament. A ZR game is essentially a layer of software that interfaces the projects written by the users or students using the IDE with the SPHERES low level code or the SPHERES embedded system (*SPHERESCore)*, which is the computerized brain of the SPHERES satellite.

A simplified version of the software hierarchy is shown in Fig. 7. The direction of the arrows indicates the direction of flow. In each autonomous SPHERE, for every control cycle, the SPHERES embedded system code sends the basic satellite telemetry information to the ZR based on its hardware and embedded system software. These comprise of the state (position, velocity, attitude quaternion and attitude rate) of all satellites operating inside the game volume, the absolute time of operations and the communication packets received from the other satellites. The ZR Game code software layer sends all of this information as well as game-specific parameters to the ZR User Code layer. The ZR User layer, which is essentially the projects programmed by students participating in ZR tournaments, uses the received information to play the game. To achieve the game objectives, the user code commands the SPHERE to set a specific state and/or sets game specific parameters using a library of API functions available for that specific game ([26]—Appendix A and B). The ZR Game Code layer receives this information from the ZR User layer and combines it with the information received from the SPHERES embedded system layer (states, time and comm. packets). Since the game code layer contains the definitions of all the API functions, the ZR Game code then updates the global game status i.e., game specific parameters. This process is indicated by the green circular arrow in Fig. 7. Based on the updated game parameters and the user commands sent from the ZR User code, the ZR Game code sends commands to the SPHERES embedded system to command the satellite's thrusters to achieve the commanded state, broadcast communication packets containing the game parameters and the self-state of the SPHERES to the other SPHERES and ping the metrology system to begin its estimation cycle. The SPHERES embedded system then initiates the physical motion of the SPHERES and the communication broadcast, in simulation or in hardware. This loop repeats itself at every control cycle of the satellite's software (set at 1 Hz frequency for the SPHERES). Additionally, the SPHERES states and state of health packets are broadcast to each other and the laptop that controls the SPHERES tests at 5 Hz.

The 'ZR Game Code' is a set of game-specific programs that are written to define the game objectives, time limits and area or volume of operation of the SPHERES satellites. Users play the game by programming their projects to achieve these objectives within the ZR User Code (as seen in the text editor or the graphical editor). When the user projects are simulated, they are done so by the SPHERES simulator along with the game code libraries and the SPHERES low level libraries (embedded system code). For hardware operations, the executable file uploaded onto the SPHERES contains the user projects, game code and SPHERES embedded system code. For any given game, the users are provided with a library of API functions that they may use within their project (within the main function or other procedures) to make the SPHERES aware of the game state, communicate with the other SPHERES and command their SPHERE to perform particular actions. The 'game code' is therefore responsible for responding to the states of the SPHERES and the user projects and accordingly, command thrusters, broadcast communication packets and update the state of the game (scores, satellite fuel, etc.). It also contains the definitions of the API functions available to the participants to command the SPHERES satellites. Together, the game code and the user projects therefore command the SPHERES (via the embedded system) to behave entirely autonomously.

The ZR game, as programmed by the game code, must meet the following criteria, developed from the lessons learned during previous instantiations of Zero Robotics tournaments and constraints of the SPHERES hardware and software:

- A game with relevance to state-of-the-art research with SPHERES, so that the work of students can contribute to future research at MIT, NASA, and other research centers.
- Each player controls one SPHERES satellite during the game, which involves two players. Games of 3 players could be possible in the future, since there are 3 SPHERES aboard the ISS.
- Each live ISS event is constrained by available ISS crew time to approximately 3 h. For effective use of resources this translates to approximately 3–5 min per match between players and approximately 15 matches per ISS session.
- The game must be easily played in 2D for ground contests on the Flat Floor Facilities at MIT or other NASA centers, but expandable to use the 3D nature of the ISS for the finals; both the 2D and 3D versions of the game must work correctly in simulation.
- Since it is not possible to manifest game pieces to the ISS for each tournament, all game items apart from the SPHERES are virtual. Games must be designed such that playing them results in SPHERES maneuvers and formation flight that are interesting to watch on the ISS.
- All matches must be bound within the physical playing area of an ISS lab.
- Due to the dynamics of the satellites, games are slower than typical arena robotics games, and collisions are not allowed. Other approaches must be used to enhance the excitement of the competition.
- The game should be such that a large percentage of the participating teams are represented on the ISS. One method of implementing this is by requiring the finalist players to be composed of alliances of multiple teams. This will enable teams to work together for the finals aboard the ISS, increasing the number of teams that participate in the finals.
- Games should be both challenging and compact, so that the game code, player code and SPHERES satellite operating system code all fit in the highly constrained flash memory available on each satellite.
- After the end of a match, each participating satellite communicates an 8-bit integer to the onboard laptop. Game scores should be such that they can be returned within these 8 bits, so that scores of each ISS and ground match can be announced immediately after completion, rather than having to wait for all the test data to be downloaded from the ISS and analyzed.

A ZR game is therefore also a full gaming environment, where the SPHERES satellites behave as robots competing or collaborating to achieve the game objectives. They not only allow students to program the SPHERES embedded systems through an indirect interface but also serve as the basis to organize educationally engaging and video-game like tournaments, where the participants get to control real satellites through the video game interface.

3.3. ZR tournaments

The 4 main phases of a generic ZR tournament are: 2D simulation phase, Flat Floor demonstration, 3D simulation phase and ISS final phase. Each phase may have one or more competitions. In each competition, students program their SPHERES satellite to play the game associated with that competition. Each competition ends with the formal submission of each team's project to control the SPHERES, following which MIT runs an automated batch simulation among all the submitted programs and declares the results. The ranks and scores may be used for elimination immediately or stored for seeding for later phases. The four phases, classified as simulation, flat floor and ISS competitions are described below.

3.3.1. Simulation competitions

The Zero Robotics programming interface provides a simulation that interprets the programs written by the students in the same way as the programs will be used in the actual SPHERES hardware. In a simulation competition, MIT runs a complete round robin among all the submitted projects for that competition, where every team competes against every other team, providing useful results for the students. The web infrastructure of ZR has an automatic batch simulation tool that allows us to run thousands of simulations by just specifying the team numbers, their associated projects and the ID of the game that they are playing—as described in Section 3. Round robins are conducted such that for every two pairs of players or programs, one match is played where the players are allocated one SPHERE each to control during the match. It is assumed that the SPHERES are identical so each pair of players plays just once, instead of twice where each controls a different SPHERE. The simulation does not replicate every aspect of the hardware; therefore, there is still a need for ground-based testing. All results, reports and animations are made available on the website for users to review and improve their software. 2D simulation competitions precede the ground competition while 3D simulation competitions precede the ISS competitions.

3.3.2. Ground competitions/demonstrations

Teams have the opportunity to run their software on the SPHERES ground hardware available on the Flat Floor facility at the MIT SSL. Plans for expanding this event to NASA Centers (initially Ames Research Center and the Jet Propulsion Laboratory) are underway. For flat floor operations, the satellites operate in 2D by floating on special air carriages that allow almost frictionless movement across the floor. The satellites can move autonomously using their thrusters, just like the ones aboard the ISS, and transmit data in real-time to the computers, which can display the motion of the satellites in the simulation environment, so that students can relate the hardware testing with their earlier simulation work. By watching

Fig. 8. A 3 Degree of Freedom (DOF) test using two SPHERES satellites on the MIT Flat Floor Facility. The onlookers are middle school students participating in the Zero Robotics Summer of Innovation Program 2010 for middle school students in the greater Boston area.

Fig. 9. Live streaming of the ISS final competition of the ZR High School tournament 2011 in an MIT Auditorium where the mentors and students from the tournament had gathered on the day of the finals to watch the live telecast. The event was hosted by 5 astronauts at MIT and 2 astronauts in the ISS.

the event webcast live, the teams have an opportunity to see the SPHERES satellites operating and learn differences between simulation and actual hardware. A flat floor competition in a ZR tournament can be seen in Fig. 8.

Feedback from the 2010 participants strongly suggested that the importance of the ground competition scores be reduced in comparison to the simulation competitions because the facilities are not as well calibrated as the ISS. Extra mass, friction and the requirement of manual assistance to help the SPHERES move caused a lot of complaints. As a result, the 2011 ground competition was held as a demonstration event only and the video footage, telemetric data and scores were available for review online on the ZR website. The Flat Floor Facility at MIT is currently being renovated so that it may be appropriate for ground competitions by 2013. Additionally, collaborations with NASA Marshall Space Flight Center and NASA Ames Research Center are being finalized such that ZR may use their flat floor facilities for ground competitions in subsequent years.

3.3.3. ISS competition

Teams that reach the final round have their programs run on the SPHERES satellites aboard the ISS with the help of astronauts. The astronauts run the final robotics game on the ISS, act as referees and interact with participating students via a live video broadcast. The final competition is a big event at MIT where all teams are invited to attend, interact with each other and watch the video broadcast from the ISS. The event will be webcast live to all participants so that teams which could not attend the event at MIT can see it remotely. Such a strong and strategic culminating event acts as an incentive to motivate students and the program therefore makes a positive impact on amateur participants. A photograph of the ISS finals of the 2011 high school tournament, hosted by Astronauts Greg Chamitoff, Richard Garriott, Leland Melville, John Grunsfeld and Jeff Hoffman at MIT can be seen in Fig. 9. The competition aboard the ISS, which is seen being streamed live, was hosted by Astronauts Don Pettit and André Kuipers.

4. Collaborative gaming in Zero Robotics

In 2011, the Zero Robotics high school tournament was themed on collaboration, to evaluate the research hypothesis that collaboration among participants improved the educational benefits gained by participants. Collaboration among participants was introduced in three ways.

4.1. Collaboration within matches

The 2011 game is focused on the topic of collaboration within competition and strives to answer the question of how teams can collaborate to achieve mission objections (crowdsourcing) while also getting ahead to win the game (exciting education). The results of the 2010 game, Helio-SPHERES, showed a lot of aggressive play, so much so that only 1 of the 10 finalist teams completed the game objectives on the ISS. All the other teams concentrated on trying to break the opponent's game and prevent them from achieving the game objectives. The 2011 game strongly incentivized communication and collaboration between the two players in the match such that playing 'together' got each more points than playing attack/defense. The 2011 game was called 'AsteroSPHERES' [34]. The theme was asteroid mining, and it was based on the premise of NASA's future missions to explore near Earth objects.

The fictional story released as a mission statement to the participants was: "Time is running out! Our planet's energy sources are dwindling and we have little time left to save the situation! BUT, not all hope is lost. Scientists have detected the presence of Helium-3 ore on two Near-Earth Asteroids, Opulens and Indigens. MIT engineers have built SPHERES satellites that can mine the Helium-3 and collect it in mining stations for Earth-transfer. The SPHERES satellites can extract the ore by spinning on (drilling) or revolving around (surface collection) the asteroids. More ore can be extracted if one satellite drills while the other collects from the surface of the same asteroid. The ore on Opulens is more enriched; however, it is protected by a layer of thick ice which has to be melted to mine it. Therefore, the mission to Opulens is much more difficult, but much more rewarding. A large mining company has leased the SPHERES satellites and embarked upon a mission to maximize the collection and delivery of the Helium-3 ore from the asteroids before their orbits take them far from Earth. The satellites can collect tools that will help their mission, but if used maliciously, can disrupt the navigation of the other. Your mission, as a team of expert strategists to the company, is to devise and implement a plan to pick up the best items, extract the Helium-3 ore, deposit it at the mining station near the asteroids and signal your success back to Earth. You will be paired up with a variety of strategist teams. If you top the charts of total ore mined for the whole mission, you will emerge as the winning team and get a large percentage of the company's profits. While you do want to get ahead of the other teams and mine more ore, it is in your best interest to collaborate to maximize ore collection. The energy future of mankind depends on you and fame and glory await you!"

In accordance with regular Zero Robotics games, each match was played by 2 SPHERES satellites controlled by opponent teams or alliances using a preloaded program (player), such that the behavior of the satellites in the matches was completely autonomous. Like the 2010 game, each player was constrained within finite resources of virtual fuel, virtual charge and code size. The virtual fuel allocation was a fixed percentage of the total SPHERES tank capacity, so virtual fuel use is directly correlated to real satellite maneuvers. Similarly, the satellites had a finite amount of power to use the tools they collect, which was *not* correlated to real battery power of the SPHERES. The satellites were not allowed to collide with each other during a match. There was an underlying collision avoidance algorithm coded within the game such that if the satellites' trajectories intersected within 20 cm of center-to-center distance in the next 10 s, then all user control was disabled and the satellites were steered in perpendicular directions to their velocity till the collision was avoided.

AsteroSPHERES consisted of three stages of 60 s each. The game had two versions: a 2D version where all the game items, objectives and behaviors were spread on the *X–Y* plane *only*, and a 3D version. Each player possessed a weak repulsor and a weak tractor, which served to repel and attract the other player, respectively. These could be used to either help or obstruct the progress of the other player, depending on the strategy chosen by each team. Participants programmed the SPHERES to play AsteroSPHERES by using available ZR game API functions within their C code.

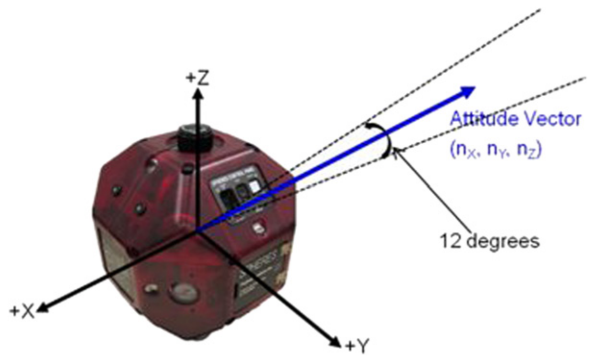

Fig. 10. Virtual attitude vector of the SPHERES, must be pointing correctly for usage of items within the game.

4.1.1. Phase one: Tool collection

Virtual tools were available to be picked up by the players: two lasers, a shield and a disruptor upgrade. A player could only pick one laser. To pick up the tools a satellite had to pass through within 5 cm of the tool's location at a velocity less than 5 cm/s. The disruptor upgrade doubled the force of the tractor and repulsor. The shield protected the satellite from the repulsor or tractor of the opponent. The laser could be used to melt the ice on Opulens (the asteroid to be mined), attack the shield of the opponent and signal mission completion back to Earth. To use any of the items, the SPHERES satellite had to be pointed in the direction of the target within a 5 degree error. The pointing direction was determined by the $-X$ face of the satellite as shown in Fig. 10. This phase did not earn points. The objective was to obtain the right tools for the strategy of Phase 2 and 3. Items that were not picked up in Phase 1 disappeared. Phase 1 in 3D is shown within the animation environment in Fig. 11. To know the status of the items within a match, the participants could program appropriate API functions into their code.

4.1.2. Phase two: Asteroid mining

Two asteroids, called Opulens and Indigens, appeared. To extract Helium-3, the players could either spin on (drilling) or revolve around the asteroids (surface collection), both of which earned points. If they collaborated on extraction operations on the same asteroid, such that one spun and one revolved, both SPHERES earned *double* the points that would be earned if extraction were done individually. For an operation to be logged as 'spinning', the satellite had to hold position within 5 cm of the asteroid location at a linear velocity less than 5 cm/s and spin as per Fig. 12. For an operation to be logged as 'revolving', the satellite had to be positioned within an annular shell of 20 cm to 40 cm within the asteroid location and revolve as per Fig. 12. The orientation of the asteroid axis was a random vector that was randomly generated for each competition but remained the same for all matches in a competition. The players could determine the orientation real-time within a match by calling an API function (See Appendix A for the full list of API functions). The idea was to teach students rotation

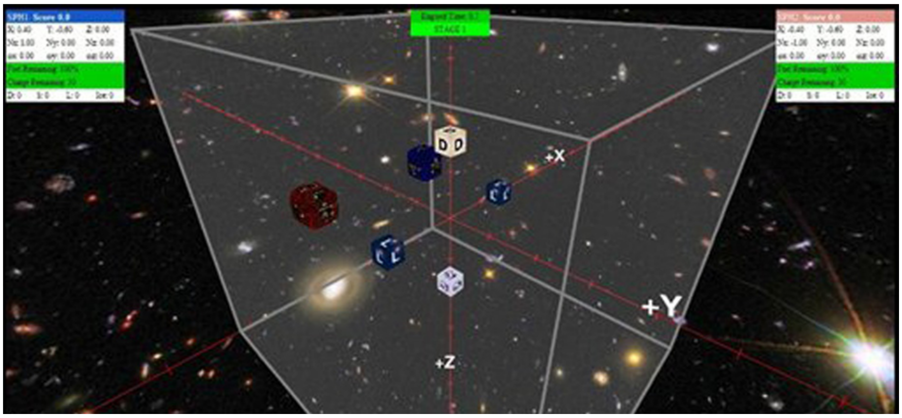

Fig. 11. Stage 1 in AsteroSPHERES3D, where L=Laser, D=Disruptor, S=Shield. The red and blue satellites indicate the initial positions of the players at the start of the game. The location of the items was different in AsteroSPHERES2D. Positions of all items and initial locations were known to the participants through the game manual. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

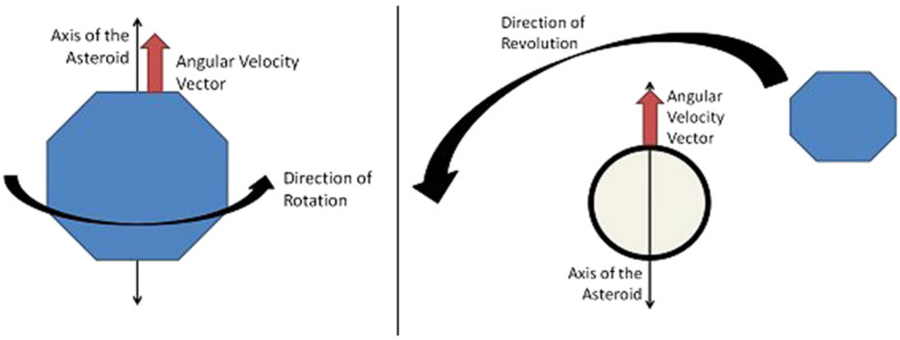

Fig. 12. Concept of 'Mining' a virtual asteroid. The left panel shows the process by which a SPHERE should be programmed to spin on the virtual asteroid. To gain maximum points that angular velocity vector must be parallel to the axis of the asteroid (axis should be perpendicular to the direction of rotation). The right panel shows the process by which a SPHERE should be programmed to revolve around the virtual asteroid. To gain maximum points the angular velocity vector about the center of revolution must be parallel to the axis of the asteroid (i.e., axis should be perpendicular to the direction of revolution).

about a generalized 3D vector and solicit a robust algorithm that was capable of achieving the goals, irrespective of random environments. The simulation settings window (Section 3.1.2) before running a simulation provided users with the ability to play with the start time of the game, phase # or items collected before any match so that they could test their algorithms within any phase of the match while programming. For formal competitions, these counters were set to zero.

Opulens had more enriched ore, i.e., worth more points, but had a layer of ice that had to be melted by shooting a laser at it (by correctly pointing toward it and calling an API function) before any extraction. Shooting the ice layer together earned more points and melted it faster. SPHERES could begin mining Opulens as soon as the ice layer melted. Indigens could be mined from the beginning of Phase 2, but earned fewer points, as it had less enriched ore.

4.1.3. Phase three: Deposit mined Ore

At the start of Phase 3 sunlight melted Opulens' ice, so both asteroids could be mined throughout this phase. In the last 10 s of the phase, two mining stations opened up. The first satellite to reach any station got that player points, but if collision avoidance was activated during this phase, both players were penalized, and the substantial avoidance maneuver disrupted their paths. 'Reaching the station' implied that the satellite held position within 5 cm of either station location at a linear velocity less than 5 cm/s. A match could end in four ways:

1. The first satellite to reach its station transmitted its "done" command, by firing a laser in a predefined direction, which ended the match.
2. Both satellites reached their stations (which earned points for *both* players, so the first one to reach the station had an incentive to let the opponent reach the station too).
3. Both satellites ran out of fuel without reaching the station.
4. 60 s elapsed in Phase 3.

The player with more points at the end of the match won and earned bonus points. The points due to the race, although also collaborative, were balanced in order to provide a competitive advantage in a largely collaborative game. Phase 3 in 3D has been shown in Fig. 13.

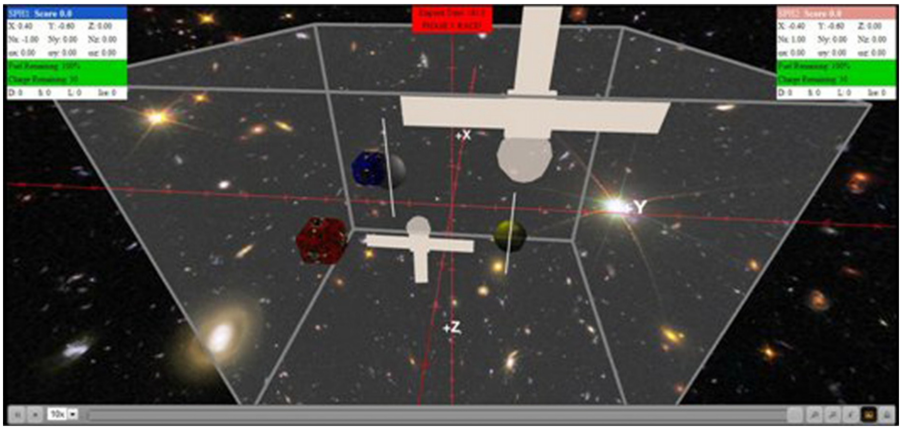

Fig. 13. Stage 3 in AsteroSPHERES3D, where the yellow transparent shell marks the position of Indigens and the black transparent shell the position of Opulens. The asteroid positions were same through Phase 2 and 3. The white lines through the asteroids indicate the orientation of the axes – randomized per match. The white T-shaped structures with the shell indicate the position of stations. The location of the asteroids and stations was different in AsteroSPHERES2D, and all were known to the participants. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

4.1.4. Match and competition scoring

The match score of a player was calculated by summing the total number of points accumulated by that player in the 180 s of the match. Points in the game could be earned, as explained before, by shooting at Opulens' ice, mining, racing or winning the match and could be lost by going outside the game volume or activating an avoidance maneuver in Phase 3. The 2011 scoring was designed such that formation flight solutions submitted could be fine resolved in terms of their relative quality. For a full analysis of scoring procedures, refer to [26], Section 4.

The scoring system and some game rules changed with every competition, such that the ISS finals were the most competitive (e.g., relatively more race points than before). Since the game was inherently collaborative, each competition was in a round robin format such that every player played every other player (players were submitted by teams or alliances—explained in Section 4.2). The competition score – to decide ranks in the competition – for any player was the sum total of the scores over all the matches played by that player. Thus, it was in the player's advantage to collaborate within each match to maximize his score rather than just beat the opponent. This also implies that the competitions were scored such that the players which could achieve the match/game objectives and maneuvers, irrespective of opponent and environmental situations, emerged higher than those who were not so capable.

4.2. Collaboration within Alliances

An important lesson learned from ZR 2010 was that there was significant loss of interest from teams that fell back after the first elimination rounds. We tackled this problem by allowing more teams (27 in 2011 as opposed to 10 in 2010) to reach the ISS finals as 9 alliances of 3 teams each. The 2011 tournament required that the 54 semi-finalists, chosen from all participating teams after the elimination rounds, form groups of 3, called 'alliances', and work together to make a common project for submission. Alliances were formed by an automatic algorithm, taking into account preferences of partnering teams and the relative seeding of teams, as will be described at the end of this section. The intent is to encourage teams to review the performances of their peers, form alliances with those they find complementary to their skill set, and work collaboratively on common projects using our online tools.

The schedule of competitions in the tournament is shown in Fig. 14: in the first two simulation competitions, one 2D (where participants played the 2D version of the game) and one 3D (where participants played the 3D version of the game), the participants competed as individual teams while in the last two competitions – one simulation and one on the ISS – they competed as alliances of three teams that submit one integrated project. As mentioned before, the website allowed each user to share his projects with other teams in the alliance such that multiple users could edit the same project, therefore making alliances with geographically separated teams possible. In fact, the EU alliances had teams that came from different countries.

The alliances were formed taking into consideration the preference of teams for partners as well as the tournament seeding of the teams. After the 3D Simulation Competition #1, the top 54 teams, ranked by the combined scores of the 2D and 3D simulation competitions, were divided into 3 tiers of 18 teams each. In the first phase, teams in the top tier ranked their preferences for alliance partners in the middle tier using a tool available on our website. Likewise, teams in the middle tier ranked their preferences for alliance partners in the bottom tier. In the second phase, MIT used this information to form the alliances. *Starting with the bottom seed* of the middle layer, each team was partnered with their first remaining preference from the bottom tier. Therefore we had a partnership between each team of the middle tier and their corresponding selection from the bottom tier.

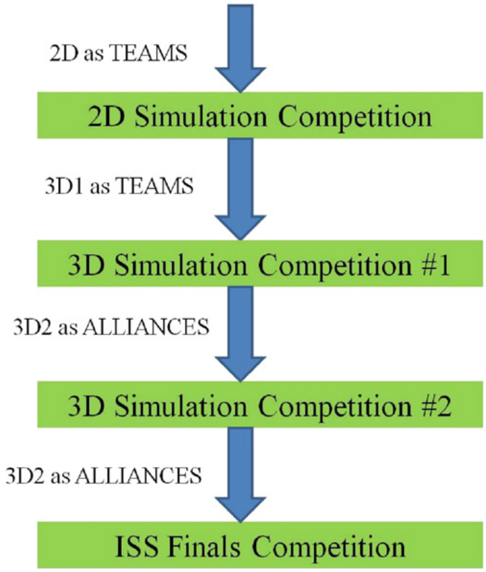

Fig. 14. Schedule of competitions within the 2011 HS Tournament. 2D competitions required participants to play the 2D version of AsteroSPHERES as the game and 3D competitions required participants to play the 3D version of AsteroSPHERES as the game. There was ~3 weeks for teams to play the game associated with the competition and submit their projects via the website for the formal simulation competition (or finally, to send to the ISS to run on space SPHERES hardware)—each blue arrow in the diagram is ~3 weeks long. All simulation competitions were essentially batch simulations of all the submitted projects by teams or alliances, run by the web administrator in the RR format after being associated with the competition's game. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

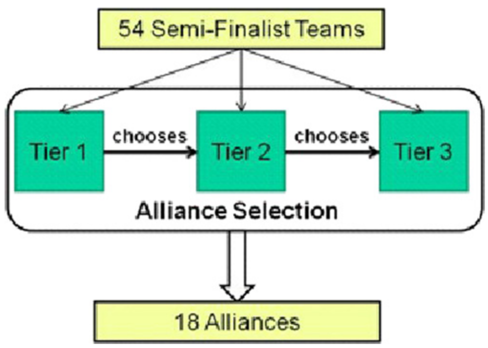

Fig. 15. Alliance selection of ZR 2011.

Similarly, *starting with the top seed of the top layer*, each was partnered with their first remaining preference from the middle tier. This resulted in an alliance comprised of one team from the top tier, its partner from the middle tier and the middle tier's partner from the bottom tier. By dividing teams into tiers and enforcing a team from each tier in an alliance with preference to the lower seeds in the second phase, we prevented the strong teams from getting stronger by partnering with only the other strong teams. The weaker teams had a chance to join forces with the stronger teams and learn from them. While all teams in the alliance could share projects and chat online with anyone who was also editing the project, only the tier 1 teams were allowed to submit projects for formal competitions. The process has been summarized in Fig. 15.

4.3. Collaboration on the community forums

The Zero Robotics website provided discussion forums for teams to communicate with each other and the game developers on the topic of programming/educational materials, brainstorming for strategies of collaboration within the matches, debating communication protocols within the limited bandwidth of data transmission between the SPHERES satellites and many other competition related interests. The forums were used extensively, with some users posting hundreds of messages. For example, AsteroSPHERES allowed the players to transmit unsigned short typed messages to the opponent player and receive the opponent's messages once every second. Teams took advantage of this facility by collaboratively coming up with elaborate communication protocols and game strategies based on the protocols. Eventually, one protocol and strategy emerged as one that more than 50% of the participants took up and followed, thus exhibiting a truly collaborative gaming environment. The challenges and project sharing tools also facilitated interaction among the teams on the website. Additionally, after every competition, MIT posted every simulated match played out in the competition on the website, in the regular animation environment so that teams could learn from their mistakes and others' exhibited behavior.

5. Impact on education

The high level goals for education and outreach using ZR [33] are to:

- *Engage students*, especially from schools that do not have funding for expensive robotics programs, in STEM activities by giving them hands-on experience with the SPHERES hardware and software.
- *Create educational materials for students* to be used both during the season and the school year for extended learning and sustained engagement.
- *Increase educator capacity and comfort in teaching STEM subject matter* by working collaboratively with certified in-school and out-of-school educators from participating schools, school districts and/or community based organizations.
- *Build critical engineering skills for students,* such as problem solving, design thought process, operations training, and team work. Ultimately we hope to inspire future scientists and engineers so that they will view working in space as "normal", and will grow up pushing the limits of engineering and space exploration.

MIT uses the unique CDIO Initiative for Engineering Education. CDIO stands for *"Conceive Design Implement Operate"* and offers an education stressing engineering fundamentals in order to create systems and products. By hands-on engagement, CDIO teaches students to appreciate the engineering process, contribute to the development of engineering products, and do so while working with an engineering organization. ZR follows the CDIO Initiative where students will *conceive* of a strategy to win the game, *design* a program using the SPHERES programming interface to demonstrate the brainstormed strategy, *implement* their projects using SPHERES hardware on the Flat Floor facilities and, using the feedback from the 3DOF environment, finally *operate* the SPHERES satellites using their projects aboard the ISS.

The different components of the educational experience delivered through Zero Robotics have been evaluated below. The intent of the following sections is to understand the observations made during the ZR program, specifically the 2011 high school season, and to design future tournaments such that educational benefits are maximized. Additionally, the effect of collaboration on educational benefits has been deduced to evaluate the hypothesis of collaborative competition is helpful to students and mentors. Note that a few results contain references to the ZR summer season conducted in both 2010 and 2011, where middle school students from handpicked schools from the Greater Boston area participated in a 5-week Zero Robotics tournament. The program was much smaller in scale than the high school one and is organized in collaboration with the Massachusetts Afterschool Partnership.

5.1. Registration status

The 2011 high school tournament received applications from 123 teams in 30 USA states. Fig. 16 shows the spread of participating schools in the US (Hawaii is not shown in the figure). Of the 24 teams that participated in 2010, 16 returned to participate in 2011, including all the 10 ISS finalists from the year. The ZR Program also expanded internationally in 2011. A select group of 22 schools from Italy, UK and Germany, handpicked under the supervision of the European Space Agency, played AsteroSPHERES on the same web platform as US schools. All the school teams participating together in the simulation competitions, however the finalists for the EU schools were selected separately. The ISS final competition was conducted separately for US and European teams and a separate champion alliance was declared for each.

5.2. Demographics

To evaluate if ZR had met its intended demographic objectives, surveys were sent to the 2010 and 2011 high school participants. In ZR 2010, 20 of the 24 participating schools (83.33%) completed the survey. There were 182 participating students with 62 mentors. The average number of students per high school was 9.1—the maximum student number was 20 and minimum was 3. The average number of mentors per team was ~3. Of the 182 students, 82.2% were male, 20.9% came from low income families, 3.1% had disabilities, and 12.15% of them had English as a second language. In ZR 2011, 47 of the 145 participating schools (31.72%) completed the survey. 90% of the students were male, 9.18% came from low income families, 3.43% had disabilities, and 13.4% of them had English as a second language. 2010 had lower responses and more minorities, as seen in Fig. 17 by the

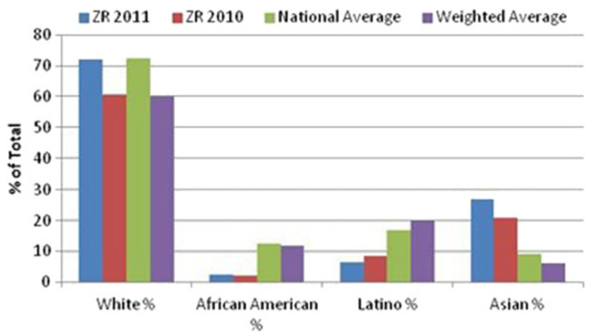

Fig. 17. Ethnic distribution of ZR 2010 and ZR 2011 HS participants, national average of the ethnic distribution of all ages in the U.S.A. and the weighted average of ethnicities in 2011 based on the statewide breakdown of demographics. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Fig. 16. Map of 123 registered US schools (2010 returning participants have been marked in blue pins). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

blue and red bars, because the schools were handpicked for a nationwide pilot, with special attention given to diversity, while 2011 was an open registration event. To compare the ZR demographics to the national demographics, data from the 2010 Census Bureau Report was used, as seen in Fig. 17 by the green bars. To measure the representativeness of minorities in ZR, we averaged the ethnic distributions of all the U.S. states and multiplied them with the fraction of ZR schools from that state in 2011 (purple bars in Fig. 17). High-school demographics of participating schools were not available. Comparison with both the national and state weighted average shows that in ZR the minorities are under-represented while Asians are over-represented by orders of magnitude. The ZR demographic numbers agree with the general trend of participation in STEM high school programs. The average national/weighted demographics need not represent the demographics of the participating districts or distributions at ages 13–17 years, which is the target age group. Also, publicity for the open registration event was primarily through NASA channels, so awareness within low-performing student districts and attractiveness of a primarily self-mentored program could have been lacking. For future years, the viral advertisement of the tournaments is planned so that more minorities can take advantage of the free, easily scalable program with extensive online tools.

A brief comparison of the performance of ZR 2011 HS teams from the U.S. with the demographic information collected through team surveys shows that performance had hardly any correlation with the female fraction in the teams, the average age of the team as determined from the distribution of students over different grades and minority fraction in the teams ($r <= 0.05$ for all). All responses in the team survey (for demographics) were sought anonymously and no identifying information was collected from user computers. All data has been reported only on an aggregate basis with no link to any one's personal identity. As a result, no individual responses or performance trends could be studied with respect to age, gender or race.

The program participation grew by 241% from 2010 to 2011. Demographic growth is calculated by the number of participants who reported to have completed the program (through feedback surveys) in 2011 vs. 2010 i.e., participated until they were eliminated by performance through a competition. The growth increase was 218% if calculated by the numbers who committed to participate in 2011 vs. 2010, as reported in their application forms. In 2010, only 51% of the applicants were chosen to participate, since the program was intended for a nationwide pilot. In 2011, all applicants were allowed to participate if they committed to the requirements of the program. To avoid selection bias, growth should be calculated using total eligible applications instead of total participants. Participants who completed the program is more reflective of the program's success than those who committed to participate, hence a growth of 241% is more representative than 218%.

In September 2010, the President's Council of Advisors on Science and Technology released a report [35] that recommends that the federal government can and should create opportunities for inspiration through individual and group experiences *outside* the classroom (recommendation #5). Recognizing the need of afterschool programs, we have partnered with the Massachusetts Afterschool Partnership (MAP) for all our middle school programs. ZR engages students through the CDIO technique and establishes STEM institutional capacity through after school programs. Unlike the HS tournaments, the MS tournaments are open to only those schools selected by MAP and funded to hire teachers to guide the students—who serve as 'Team Mentors'. Additionally, each school has been exclusively supported by one MIT undergraduate mentor. The MIT mentors ensured that the students individually and as a team made sufficient progress with programming the SPHERES in order to complete the game successfully.

From the feedback surveys conducted after the middle school program in 2010, the statistics show success in achievement of our goals. There were over 200 middle school participants from 10 schools in the greater Boston area. 84% came from low-income families, 81% from ethnic minorities, 54% were female and 75% from low-performing school districts. The youngest participant was a rising 4th grader! All ten programs had a retention rate of 88% or greater and a daily attendance rate of 90% or greater. Due to funding limitations, the middle school program scaled down in 2011 and only 5 of the best 2010 schools participated. Of the 68 students in 2011, 31% were female, 79% came from low-performing school districts, 9% were diagnosed with learning disabilities and 10% were English Language Learners. The MS teams showed far more diversity than the HS teams because MAP selected the schools to uphold its objective of education for all.

5.3. Educational quality

The quality of STEM education delivered by ZR has been measured in two ways: by analyzing improvements in game scores as the tournament progressed, and by using post-tournament surveys to obtain firsthand participant feedback regarding the educational impact of the program. The 2010 surveys were significantly qualitative since it was a pilot, and the descriptive feedback was intended to help design the open registration website for 2011 and further. In contrast, the 2011 surveys were quantitative in nature with text space for providing optional written feedback. The findings are presented below.

There were two surveys- one team and one individual- both entirely online and available to all participants. Each team was requested to submit one response to the team survey, filled out preferably by a team mentor. Each student participant was requested to submit one response to the individual survey. 240 mentors and students responded to the individual survey (out of 1274 open-registration participants from the US and $\sim$100 from EU) and 47 teams responded to the team survey (out of the 145 teams whose applications were accepted and 110 teams who participated in at least one competition in the tournament). In the rest of this section, individual or

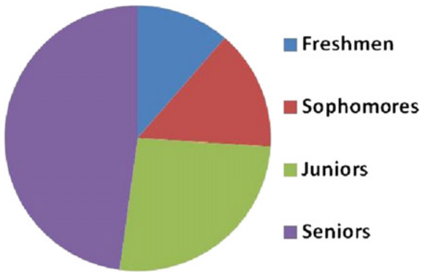

Fig. 18. Distribution of students among the 4 HS classes based on the sample of students who responded to the survey among the population that participated.

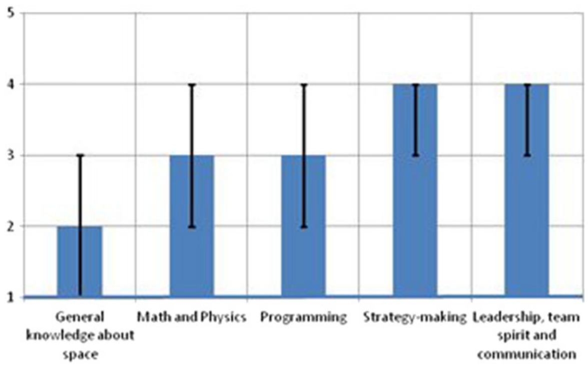

Fig. 19. Median of responses to: "On a scale of 1 (no improvement) to 5 (significant improvement), please rate how the ZR Spheres Challenge improved your skills in the 5 mentioned areas". (Error bars indicate the inter-quartile range) The horizontal blue line marks the neutral level. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

student survey responses refer to participants *speaking about themselves*. Mentor or team responses refer to mentors speaking about *their overall team*. While all participants were reminded multiple times to respond to the surveys and incentives in the form of merchandise goodies were provided, potential for 'adverse selection' of respondents was ever present, especially since the sampled number was less than 50% of those who were potentially affected. The possibility that this may have biased conclusions drawn from the surveys to some extent cannot be completely ruled out.

The distribution of the participating grades in the HS program, shown in Fig. 18, indicates that nearly 50% of the students are rising college freshmen i.e., only about half the current participants will be able to continue the program next year. The 2011 individual survey asked the students to rate the improvement of their skills in five different target areas. These target areas were chosen in a way as to measure as fully as possible the educational impact of the ZR program along the six primary objectives for federal STEM investment (as articulated in the 2010 Federal STEM education Portfolio Report [36]): Engagement, institutional capacity, learning, leadership, STEM degrees and careers.

Fig. 19 shows that the participants found their leadership, team-building and strategy-making skills the most improved, followed by programming, math and physics. The 2011 game was strategy-intensive and designed with the intent of incentivizing learning and achievement through collaboration and strategy. The idea was to get the students excited through peer-based learning techniques. This would potentially provide impetus to even the least STEM inclined to start off on improving their basic CS-STEM and teamwork/leadership skills. The survey results (through ordinal data analysis) also show that more than 75% of the participants reported math, physics and programming improvements (Fig. 19) and more than 90% reported leadership and strategy improvements (ratified by calculating the 90th percentile in Fig. 19 data).

Students were also asked the question *"How much has your inclination towards STEM increased due to the program?"* on a Likert scale (1=Not increased at all, 2=Not much, 3=A noticeable amount, 4=Significantly, 5=I am now certain of a career in STEM) to which their median response was 3. 89% of the participants in 2011 reported a measurable increase in STEM interest due to the program based on this question, and 15% declared, *"I am now certain of a career in STEM!"*. The increase in STEM inclination yields a weak correlation (Pearson's correlation coefficient 'r'=0.26) with the average number of hours that the participant reportedly spent on the program.

To guard against self-assessment bias, the responses of mentors that were relevant to their team's STEM improvement metric were taken into account (full results shown in Fig. 20). Mentor assessment shows 'satisfactory increase' in programming and leadership abilities, with low range, and a nearly satisfactory increase in STEM inclination. Since 75% of the mentor responses lie above the neutral line (indicating 'No Change' due to the program), ZR is concluded to have significantly met the federal primary STEM objectives.

Mentors gave ratings of 'improvement in programming' that were about the same as those the students gave, but their ratings of improvement in leadership and interest in STEM fields were lower. The comparison of responses is shown in Fig. 21. 85% of the mentors (speaking about their team) and 86% of the students (speaking

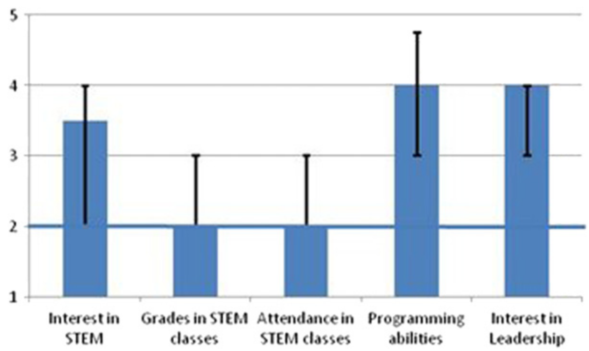

Fig. 20. Median of responses to "Please rate the students in the team on the following academic/education indicators compared to before the SPHERES Challenge 2011 where 0=Have no information, 1=Decrease, 2=No change, 3=Small but noticeable change, 4=Satisfactory increase, 5=Very significant increase". Error bars indicate the inter-quartile range of responses. The horizontal blue line marks the neutral level. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

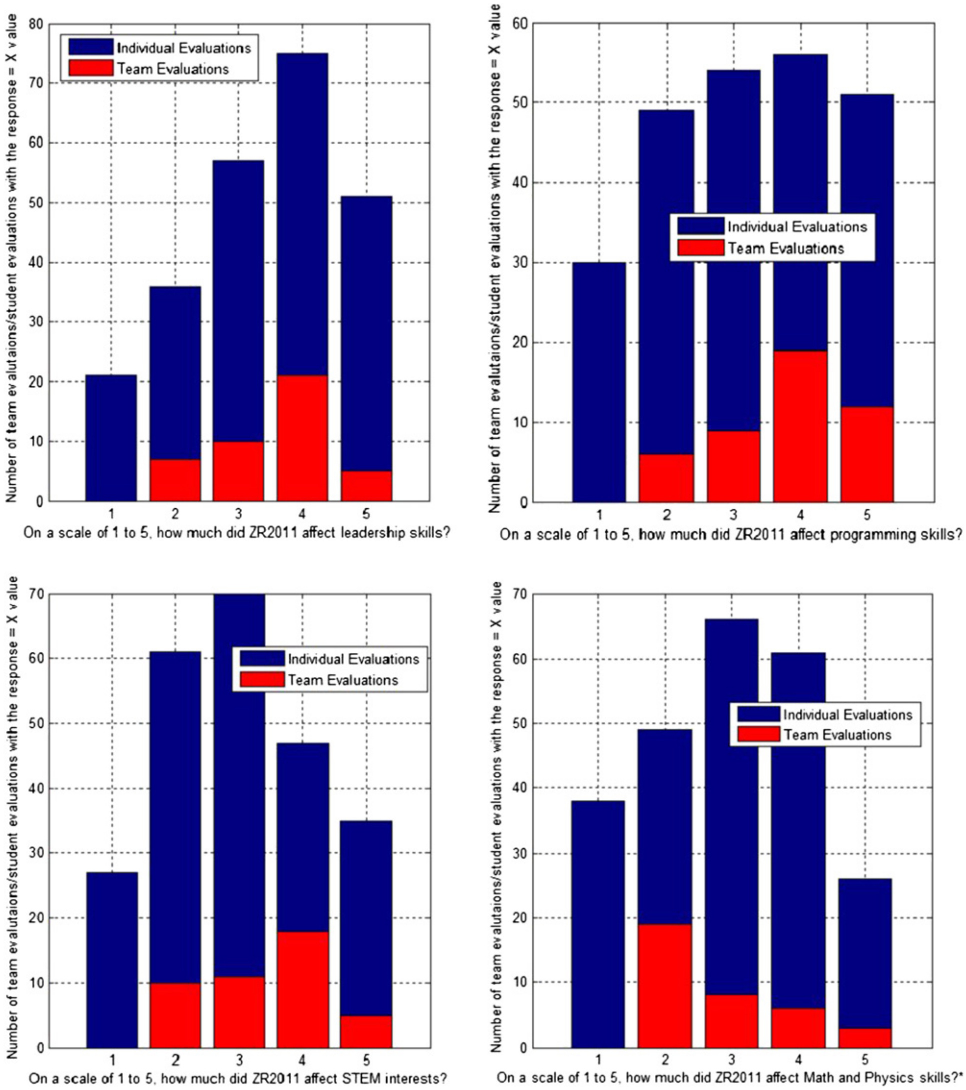

Fig. 21. Histograms of responses to team (red) and individual (blue) surveys, on the effect of ZR 2011 on (roughly) the same 21st century skills [6]. The neutral response (indicating "No Change due to the ZR Program") was (1, 2) for the (individual, team) survey, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

about themselves) reported a positive increase in their programming skills. 89% of the students but only 77% of the mentors reported a positive increase in the team's leadership skills. Similarly, 88% of the students but only 73% of the mentors reported a positive improvement in team's STEM inclination. The neutral response along the Likert scale was at 1 for the individual evaluations and 2 for the team evaluations, which rates program improvement evaluation on a 5-point scale for individual surveys but only a 4 point scale for the team surveys.

The bottom, right panel of Fig. 21 evaluates the math and physics skills improvement measured in two *different* ways. The individual survey asked by how much students perceived their skills to have improved, while the team survey asked by how much the mentors knew student grades to have improved. The figure shows that students reported a high individual improvement in math and physics skills (86% of them reported positive results). Their grades in those classes, however, showed only a small improvement ($<25\%$ showed positive results). Assuming that the students are not greatly exaggerating their improvements—a fair assumption given the correlation in the other 3 panels in Fig. 21—the net effect of these interventions is not immediately apparent in their school work. However, such indicators should be measured over the arc of their educational careers, perhaps every few years, to truly assess the long-term benefits of ZR.

One important lesson learned in the analysis of mentor and student feedback is that the scale of comparable responses in a survey should be kept the same, so that comparison across questions is possible; the language and description of questions that seek the same answers should as nearly as possible be identical. The 2011 survey was a pilot evaluation of a new observational study and the process has proven to be a valuable learning experience for us in how to ask the right questions in the right way.

Mentors of alumni teams were asked the following question: "If you participated in last year's SPHERES Challenge 2010, please check the year (2010 or 2011) that you felt contributed more to the improving education indicators below…" 55% of the alumni teams reported that 2011 contributed more (between the two years of participation) towards increasing interest in STEM and Leadership and most of the others reported that they contributed equally. Also, 65% of alumni team members reported that 2011 contributed more to their programming abilities, and again, most of the others reported that the two years contributed equally. Overall, this suggests s that ZR's has contribution has improved educationally. Furthermore, 89% of the teams that responded to the survey said that they would participate again in 2012.

To evaluate program satisfaction, participation motivation was studied and the program evaluated on grounds of achievement of its perceived motivational factors. Participants were asked to rate the reasons why they participated in the ZR tournament on a scale of 1 to 5 and the top 6 results are shown in Fig. 22. MIT's name, the intricate engagement of the program with space, and programming skill-seeking emerged at the top of the motivational factors. The choice of the factor options for the survey was based on the vision for ZR's foundation and feedback from participants in the 2009 and 2010 pilot programs. For example, a 2010 mentor reported, *"We do not have a computer programming class at our school so this was a great activity and teachable time for students that were interested in programming to gain experience and accomplish a goal"*. Another mentor had said "*This was one of the coolest projects I've been involved with. The fact that we were working on code that might eventually fly on the ISS was a very compelling motivator for the kids.* "

Significant amount of effort has been invested in ensuring that the value returned to participants for each motivational factor is high to maximize participant satisfaction. To allow teams the full MIT experience, the 2011 ISS finals event was held in a large MIT auditorium where ALL teams were invited. The event was hosted by 5 astronauts in attendance while the competition streamed in live from the ISS, hosted by 2 astronauts in space. Teams were able to meet their competitors and collaborators and interact with the MIT staff, all of whom they had met only over the ZR web interface. Attendance surveys showed that 245 participants (including 16 non-finalists) attended the event from 19 teams. For remote participants, the event was webcast live and screened live on NASA TV for over 6 h on January 23rd, 2012. The 12 finalist alliances, comprising 36 teams, had their programs sent up to the ISS. Students saw 'astronauts run something they created' and successfully 'controlled robots in space'. 36 teams of the 145 that submitted an application (25%) and of the 91 teams that submitted a project to the tournament (40%) saw their motivational factors met. Programming knowledge objectives were met, as shown in Fig. 20, wherein mentors indicated it to be the highest skill gained due to the program. Finally, to understand the value returned to the factor 'I like playing games', the students were asked an independent question: *"Compared to other video games/programming games you have played, how hard did you find Astero-SPHERES?"* where the response options were: 1 = Fairly easy, I'd have liked harder challenges, 2 = Difficult in the beginning, but was got a bit boring toward the end, 3 = Challenging and engaging all through, 4 = Too difficult for me to compete confidently. The median and mode of the responses peaked at 3 (Challenging and engaging all through). Overall, the program met its motivational objectives satisfactorily and we now have a baseline in place to measure subsequent changes to the motivation and its achievement in the future years.

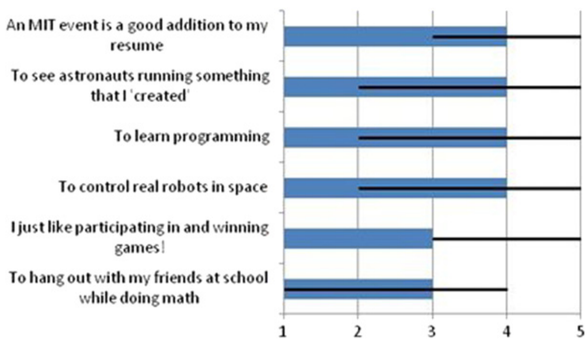

Fig. 22. Median of responses to "Why did you participate in the SPHERES Challenge? On a scale of 1 (hardly a motivator) to 5 (significant motivator) please rate how much the following served as reasons". Error bars indicate the inter-quartile range of responses.

Optional descriptive feedback submitted by mentors and students indicated that apart from the factors in Fig. 22, ZR appealed to them because it was a practical hands-on application of HS math and physics. A mentor provided the following feedback, "*I normally mentor programming contests with the students and this was different. The problem was more "real-world" and involved more strategy than just problem solving."* This tied in very well with ZR's founding principles and the paper objective which is to provide accessible, real-world CS-STEM education to students.

Lastly, the performance of teams and alliances in competitions within the tournament and how they improved over time is an important metric of the educational quality of the program. In future years, to improve value to students, the projects of the top performing teams in a competition may be published on the ZR website (with permission from the authors) so that other teams may learn from them and build on existing know-how. Pre- and post-tests were administered during the Zero Robotics Summer Program 2011 for MS students from the greater Boston area. Quantitative evaluation of results show that students' interest and engagement in the STEM fields increased as a result of participating in the program ([26] Appendix C).

5.4. Effect of collaboration

One of the objectives of the 2011 tournament structure and game was to introduce various elements of collaboration and understand their effects on STEM education. It was hard to measure the independent effects of each collaborative factor without a tightly constrained human experiment. Since ZR is primarily an educational effort in which participants are meant to enjoy a fair game, the effect of collaboration on STEM education is measured using multivariate quasi-experimental analysis on passively observed/studied data. For a full description of research methods utilized and justifications, refer to [26], Section 4. Quasi-experiments [37,38] are distinguished from true experiments primarily by the lack of random assignment of subjects to experimental and control groups and sometimes, the lack of control groups (as is the ZR case).

The results from the team-based and alliance-based results from the tournament indicate that alliances of teams showed higher average scores than individual teams, demonstrating the importance of collaboration environment #2. As seen in Fig. 23, the mean score among all the teams has improved significantly after grouping the teams as alliances. To mitigate the effect of selection bias on experimental validity, only teams that participated as alliances in the 3D#2 Competition were chosen for analysis in the 3D#1. See Fig. 14 for the sequence of competitions. The mean score of the team competition (3D#1) was 9.1 (standard deviation of 5.6) and the mean score of the alliance competition (3D#2) was 14.6 (standard deviation of 4.6).

The mean of the alliance scores is more than one standard deviation greater than the mean of the team scores. However, the scores are not normally distributed (by the Kolmogorov Smirnov test), hence a t-test could not be used to calculate the differences. The interpretation of this difference in scores is further complicated by possible learning over the three week interval as well as minor modifications in game rules between the competitions. For instance, while it was entirely possible to have a perfect score in the first competition by programming a perfectly collaborating, strategic revolve maneuver around Opulens, getting a perfect score in the second competition additionally required a perfectly timed trajectory and a perfect maneuver to dock to the mining station.

The score distributions of the 2D and 3D#1 Competitions were compared to find the effects of the learning period and game rule changes. Both competitions had a 3 week period of preparation/programming (schedule in Fig. 14), team participation and modification of game rules. While the competitions received 88 and 91 submissions respectively, only the 70 teams that participated in 3D#1 were chosen for analysis in 2D. The mean 2D score was 6.2, standard deviation 4.78, and the mean 3D score was 7.83, standard deviation 5.6 (Fig. 24). From the figure, it is easy to visually interpret that the improvement in the mean between 2D and 3D#1 is far

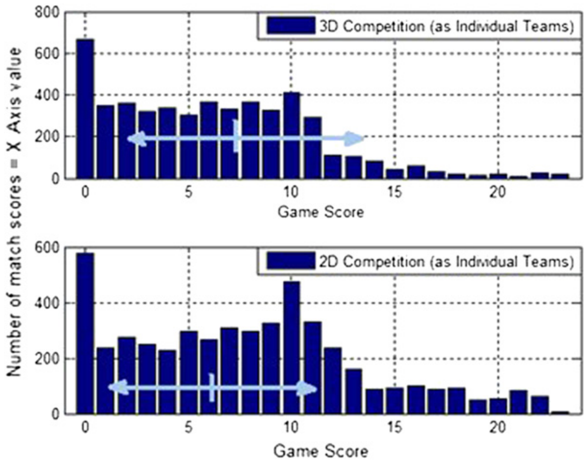

Fig. 24. Comparison of the 3D#1 with the 2D scores (both played as teams). The 3D#1 competition is the same as that shown in the left panel of Fig. 23, but only those (70) teams that played both 2D and 3D#1 were chosen for analysis. The mean and standard deviation of each set is shown in the figure.

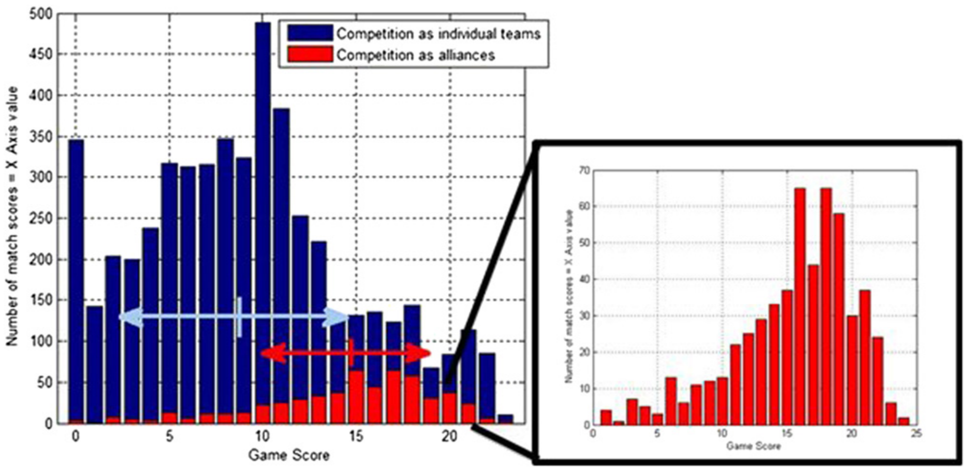

Fig. 23. Comparison of score distributions with and without alliance-based collaboration. The blue bars are the scores of teams in 3D#1. The red bars are the scores for alliances in a separate competition, 3D#2. The blue histogram contains 4095 round robin match scores, played between every pair of the 91 teams. The 72 highest teams were formed into 24 alliances, of 3 teams each. Thus, there were 8190 (blue) and 276 (red) matches in 3D#2. The mean and standard deviation of each set is shown in the figure. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

less than the improvement in mean from 3D#1 to 3D#2. It is therefore reasonable to conclude that a larger share of improvement in game scores when alliances were introduced was due to the existence of the alliance variable rather than the learning and game rule modification variables. This conclusion, however, assumes that the combined effect of game rule modifications and participant maturation between the two sets of competitions is equivalent. This assumption is cannot be verified because neither set is quantifiable and both are unrelated changes. No other control was available for this observational study.

Time-series analysis, which is the analysis of the changes of a variable in time sometimes with the use of another time series to counter the effect of a third possibly confusing variable, is a method of analyzing quasi-experiments. Time series analysis was applied to the variable *competition score per teams* to assess the effects of collaboration in alliances on the scores of teams. There were 4 competitions in the 2011 tournament (Fig. 14). Registered teams participated in the 2D Competition (2D) and then the 3D#1 Competition (3D#1). Team scores from both competitions were weighted at a pre-declared ratio of 1:3 and the 72 highest scoring teams were eligible for the 3D#2 Competition (3D#2). 3D#2 required teams to compete as alliances of 3 teams each chosen from 3 different "tiers" of performance. Each alliance thus had a set of teams that had performed very diversely in the 2D and 3D#1 competitions. For the purpose of time series analysis, only 54 teams that participated in ALL 3 competitions above were considered. Each team played once against every other team in both 2D and 3D#1. Thus, every team played 53 matches in each of the two competitions.

Each alliance played against each of the 17 other alliances in 3D#2. We calculated the average match score of each original team (grouped vertically by the alliances they would later join) in the 2D and 3D#1 competitions using Eq. (1). and plotted them in Fig. 25 (red and blue error bars respectively). The mean score calculated by averaging over their future alliance, i.e., Eq. (2)., is plotted in blue and red asterisks. The average match score of each alliance in the 3D#2 competition, i.e., using Eq. (1). with alliances instead of teams, is plotted as black asterisks.

$$\text{averageMatchScore(team, competition)} = \frac{\sum_{\text{match}}^{\text{matches in comp}} \text{matchScore(team)}}{\text{number\_of\_Matches\_in\_Comp}} \quad (1)$$

$$\text{meanAllianceScoreOverTeams(alliance, competition)} = \frac{\sum_{\text{team}}^{\text{3 teams in alliance}} \text{averageMatchScore(team, competition)}}{3} \quad (2)$$

The mean score per match over all matches in each competition is plotted using a broken line—calculated by summing over Eq. (2)., by competition. As discussed previously, the overall average increased from 3D#1 when there were no alliances to 3D#2 where teams played as alliances. Moreover, ALL teams showed an improvement by participating as alliances as seen in Fig. 25. Since there was a 3 week learning gap and modifications in game rules between the two competitions, those changes could have contributed to the improvement.

The competitions 2D, 3D#1 and 3D#2 address the same problem in satellite (SPHERES) programming among the same subjects, on a game with the same structure in each case. The competitions occur three weeks apart and the placement of virtual objects in them is slightly different which affects the optimal strategy. Because of those (incomplete) similarities we use the difference between 2D and 3D#1 scores as a partial control for the difference between 3D#1 and 3D#2 scores. That difference was 3.4 points per original team over all the alliances (Eq. (3)). The

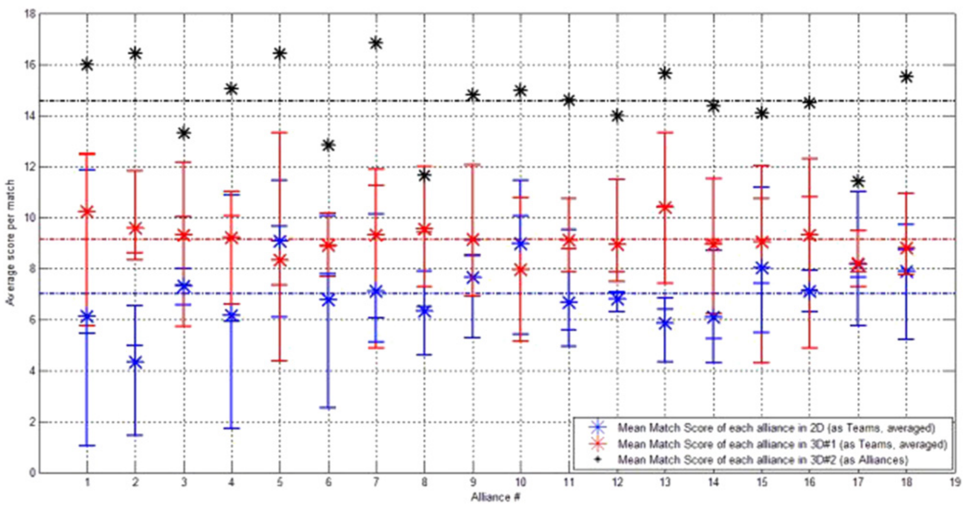

Fig. 25. Alliances whose component teams participated in all 3 competitions (2D as Teams, 3D#1 as Teams and 3D#2 as Alliances) plotted against the average score of the alliance per match in the 3 competitions. For each asterisk, the error bar's horizontal line indicates the mean score of the component teams of the alliance for that competition; for 2D and 3D#1. The horizontal dotted line indicates the average score for that competition over all teams and alliances. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

same calculation was done for Figs. 23 and 24.

$$\text{ScoreImprovement(team)} = \text{average Match Score (team's Alliance, 3D\#2)} - 2*\text{average Match Score(team, 3D\#1)} + \text{average Match Score(team, 2D)} \quad (3)$$

Fig. 25 shows the broad range of 2D and 3D#1 scores of teams that came together as alliances—this is due to the tier system of alliance selection. As explained above, each alliance had one team from each of the three tiers of performance. Tier 3 teams showed the greatest improvement in performance from 3D#1 to 3D#2. These teams had the maximum opportunity to improve and it appears, the higher performing teams in their alliance helped them learn and improve quite rapidly.

A similar time-series analysis was done using the relative ranks of the 54 teams, based on their total score in all the matches in that competition. Using ranks is convenient because Tier 1 teams have a smaller opportunity to improve their scores compared to others (their scores were already closer to the theoretical maximum of 23). Using ranks somewhat mitigated the risk of statistical regression [38,37]. As explained in Section 4.1, in any competition for the 2011 tournament, the team that received the maximum *total* score in a competition (i.e., the summation of all its scores over all its matches in the round robin competition) received the topmost rank. Therefore, the average match score of a team/alliance as plotted in Fig. 25 determined their rank in the competition. Fig. 26 shows the improvement in rank for each original team from 3D#1 to 3D#2. In the 3D#2, the team is given the rank of its alliance's performance. Performance range is calculated as the 1-norm, the weighted range of each original team's score with respect to the average score its alliance. This was calculated using Eq. (2) where averageMatchScore and meanAllianceScoreOverTeams is given by Eqs. (1) and (2)

$$1-\text{norm(team)} = 0.75*|\text{averageMatchScore(team, 3D\#1)} - \text{meanAllianceScoreOverTeams(alliance, 3D\#1)}| + 0.25*|\text{averageMatchScore(team, 2D)} - \text{meanAllianceScoreOverTeams(alliance, 2D)}| \quad (4)$$

The scatter plot in Fig. 26 is U-shaped because we use the absolute value in Eq. (2). Tier 2 teams have average match scores closest to the mean alliance score and therefore the lowest difference under Eq. (2). By contrast, Tier 1 teams and Tier3, the extreme performers in their alliance have the largest distances from their alliance's mean score. The plot affirms that Tier 3 showed the maximum rank improvement and a Pearson correlation of 86% with the 1-norm range. This supports the general conclusion that the tier-based system of alliance selection used in the program was effective in bringing the competition spotlight on Tier 3 teams. On the other hand, negative correlations of Tier 1 and Tier 2 teams show that the diversity in recruiting fostered by the alliance formation protocol did not help them climb in rank.

While Fig. 26 showed that the *performance* of Tier 3 teams improved, it is important to examine whether this improved performance as an alliance was reflected their individual learning as a team. After the tournament, teams that competed as an alliance in the semi-finals and later were asked, *"How much of the alliance code did your team contribute?"* with the options of 5 = Our team did all the alliance work, 4 = Most of the contribution was ours, 3 = Almost exactly 1/3rd of the work, 2 = Much less than 1/3rd of the work, 1 = Our team did not contribute to any alliance work. The range of responses of the teams (1 to 5) was normalized to (0 to 1). The average self-assessed contribution to the 3D#2 project of the Tier (1, 2, 3) was (0.909, 0.361 0.477) respectively. Stronger teams evidently felt that they contributed far more to alliance software and performance than the weaker teams did.

Fig. 27 shows the (second level) difference between two improvements in an alliance's average score per match (Eq. (3)); the improvements between 3D#2 and 3D#1, less the improvement between 3D#1 and 2D. It is a

Fig. 26. Scatter plot of the drop in rank (i.e., performance improvement) of 54 teams between the 3D#1 and 3D#2 competitions vs. the absolute range in their 2D and 3D#1 scores with respect to their alliance's mean, grouped by their alliance Tier Number. The Pearson's correlation coefficient for each tier's scatter plot is indicated in parentheses.

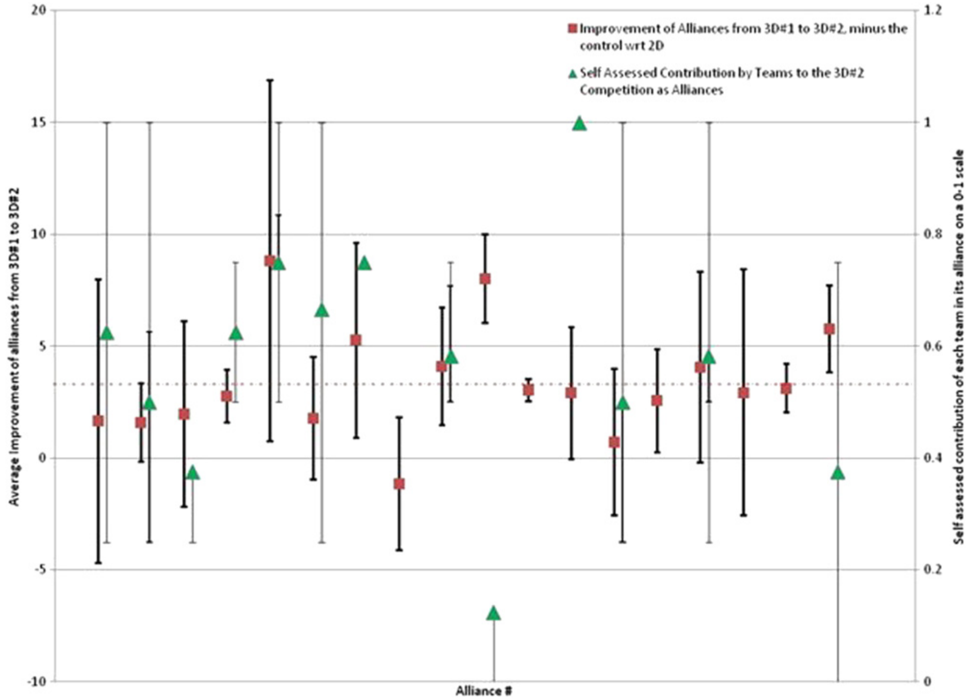

Fig. 27. Change in average score per match from the 3D#1 competition (as Teams) to the 3D#2 competition (as Alliances) minus the control [2D minus 3D#1], to account for student learning and game change between the 2 competitions. The error bars indicate the standard deviation of the team averages with respect to the alliance averages (red squares) of match scores. The self assessed contribution of teams to their alliance's project (mean marked as green triangles, individual team responses, as bars) has been plotted on the secondary axis. The overall increase in the mean score, over the control was 3.4 points (horizontal red dotted line). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

rough measure of how much more they improved above their performance as separate teams before the alliances were formed. The red squares in Fig. 27 indicate the averageImprovement by alliance, calculated by Eq. (3).

$$\text{AverageImprovment(alliance)} = \text{averageMatchScore(alliance,}3D\#2) - \text{meanAllianceScoreOverTeams(alliance,}3D\#1) - [\text{meanAllianceScoreOverTeams(alliance,}3D\#1) - \text{meanAllianceScoreOverTeams(alliance,}2D)] \quad (5)$$

Fig. 27 also plots the self-assessed contribution of each team in the alliance on a scale of 0–1 beside the actual improvements in score. The averaged self-assessed contribution over all team responses in an alliance is marked by a green triangle. Since some alliances had no responses from any of their teams, there are alliances without green triangles in Fig. 27. The individual team responses (if received) are marked by horizontal bars about the triangles.

The ideal average contribution per alliance on a 0–1 scale should have been 0.33, irrespective of the spread of individual team contribution. The average of the self-assessed contribution was higher than that. This indicates that among the alliances in which all teams responded, many teams must have assessed their contribution to be greater than it may have been. Fig. 27 shows a large overall variation in the estimated contribution by teams to their alliance. The variation was not correlated ($r = -0.01$) with the improvement in alliance performance (Eq. (3)) and weakly and negatively correlated ($r = -0.3$) with the improvement in team scores (Eq. (3)). The latter correlation was not what the program intended—we had hoped the better scores would reflect a uniform contribution Instead, Tier 3 teams improved the most but claimed to have contributed the least. Conversely, Tier 1 teams improved least and claimed to have contributed most.

The analysis shows that performance scores of alliances alone are not enough to assess the educational value delivered to the teams. From survey responses, it appears that alliance formation limited the contribution of the weaker teams and slightly reduced the relative ranks of the stronger teams. This observation could potentially be attributed to demoralization bias—weaker teams may have been assigned less interesting or more menial work within their alliance and, as a consequence, felt they did not learn or contribute enough.

To investigate the effect that the special tier-based alliance selection method (Section 4.2), had on the improvement of teams' performances, team performance diversity (Eq. (1) divided by averageMatchScore(team, 3D#1) from Eq. (1)) was correlated against % improvement of match scores after alliances were formed—Fig. 28. The correlation between the two variables is weak and positive, indicating that although the large difference in capabilities of the alliance teams correlated positively with the improvement of the teams' performance, it was weak ($r = 0.34$) and tells nothing about causes. On the other hand, improvement in team ranks (X axis in Fig. 26) correlated moderately and *negatively* with their average weighted 2D+3D#1 scores. In fact the correlation

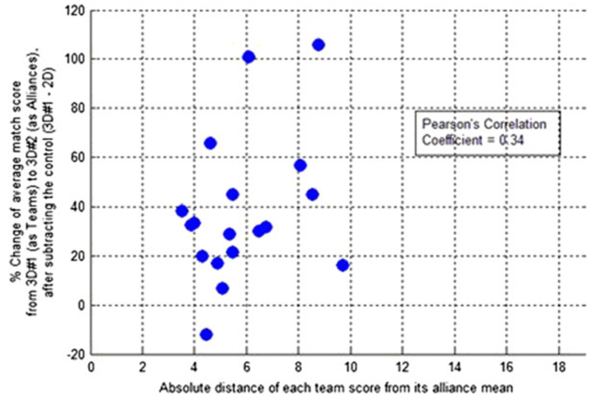

Fig. 28. Scatter plot of the 1-norm range (Eq. (2)) of 2D and 3D#1 scores of each team in an alliance about the alliance mean score vs. the fractional improvement in average match score of each team from 3D#1 to 3D#2.

coefficient for Tier 1 teams alone was −0.4 and lower than the overall coefficient of −0.22. The individual capability of teams was not enough to help them improve. A finely balanced technique of inducing diversity in alliances without diluting their performances is therefore important.

While some respondents found the diplomatic collaborations within alliances interesting, found good ideas through them and agreed that they increased the overall STEM participation, others found it disappointing to remotely get in touch with teams that they had not worked with before and resolve differences of opinion. A mentor from the tournament said, "*It has been very interesting to work with another team, but I think the third team cannot add a significant value to the alliance. Moreover, a team that has performed badly in the qualifying phase can access to the finals if allied with a skillful team.*"

In future years, ZR will seek to design games that evolve significantly between the non-alliance and alliance competitions (e.g., 3D#1 and 3D#2 in 2011)so that the three teams in an alliance have enough remaining work to divide efficiently among themselves. No team should feel left out. The web infrastructure will require tools to promote equal contribution by all teams in an alliance. In the 2011 web interface, while all teams within an alliance could share projects with each other and use ZR's instant messaging tool to chat with each other when editing projects, the submissions tool allowed only Tier 1 teams to submit the alliance's project to a formal competition. Small details like these have been known to create a sense of alienation in Tier 2 and Tier 3 teams, as revealed in some of the essay surveys. Additionally, a re-evaluation of the alliance selection mechanism may also be needed. Tier 1 teams stronger expressed the desire to partner another Tier 1 team in an alliance, especially since discussion forums had forged friendships between already motivated teams.

The negative correlation of rank improvement of Tier 1 and Tier 2 teams as indicated in Fig. 26 and the low correlation between a team's performance variability as part of an alliance and the same team's overall performance improvement provides an incentive to reconsider the idea of teaming diverse performing groups. While the formation of alliances has apparently increased the overall performance of the group and received wide approval among the participants, there is room for improvement through the revision of the *methodology* of grouping teams into alliances.

While 63% of the survey respondents found the collaborative game challenging and engaging, and even intimidating, essay responses seem to indicate the way collaboration was implemented in the game and tournament was partially the reason why 33.8% found the game "*Difficult in the beginning, but got a bit boring toward the end*". Some students wanted a more adversarial game and many participants wanted more substance in the game after teams grouped up as alliances, so that each team would have something extra to do. It is important to note all respondents unanimously expressed that the collaborative nature of the tournaments should be retained to some capacity. From this feedback, the lessons learned for in-game collaboration are that while collaboration was well received as an objective, the game should have more adversarial components than just a finale race.

There were several web-based collaboration tools available to the participants in 2011 such as project sharing tools, an project instant messaging (IM) system among all users among who a project is shared, informal challenges such that teams could play matches against each other outside of formal competitions and a discussion forum. The participants were asked to rate the project sharing feature and IM chat features on a 5-point Likert scale and the results are shown in Fig. 29. While project sharing was very well received and chat room feedback shows that up to 50% of the population might not have known about the chat application. This is because the chat application, like challenges, was released well halfway into the tournament and did not receive attention during the kickoff introductions. Since the website is now well developed, it is expected that the features will be well advertised next year.

To understand the usage of discussion forums, the scores of teams in the 3D#1 competition was correlated with the number of forum posts by users of the team. This competition was chosen because it was the last one before forming alliances and we wanted to make any conclusions drawn independent of the alliance variable. The Pearson coefficient (r) was 0.37 which indicates a

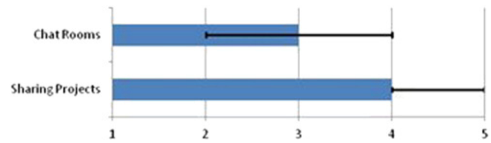

Fig. 29. Median of responses to the individual survey question: "Please check that which applies to each of the collaborative features below". The response options were 1 = Found it annoying, 2 = Had problems using it, 3 = Did not notice/use this at all, 4 = Used a lot but would like this improved, 5 = Found this extremely helpful. Error bars indicate the inter-quartile range of responses.

moderate positive correlation. The scatter plot of the data values is shown in Fig. 30. It must be noted that, visually, the data seems to be bounded in a quadratic curve and is not linearly arranged i.e., very low performers hardly participated in the forums, very intense forum participants had high scores but there were high performers with relatively low participation. This analysis is a correlation and does not imply causation. The observed trend implies that students who were participatory and frequent at the forums tended to do well—more collaboration, better results. The trend could have been further strengthened due to the collaborative nature of the game. More forthcoming teams had the strategic advantage of interfacing with other teams to make a block of successful collaborators (e.g., protocols introduced in Section 4.3 and available in greater detail in [26] Section 5) while the quieter teams either efficiently programmed the strategies being discussed or did not invest effort in strategizing or programming. Overall, the message board system was very educationally popular (as indicated by the essay-type feedback) and logged a total of 5150 messages by 164 unique users in the entire tournament period.

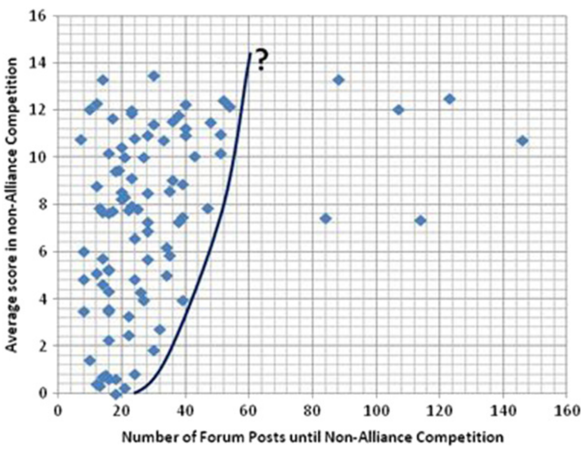

Fig. 30. Scatter plot of the number of posts made by a team on the website discussion forums of before the submission deadline of a competition versus the average match score obtained by that team in the same competition. Correlation coefficient (r)=0.37, quadratic trend seen.

To measure the value the participants felt they gained through the various features provided by the ZR program, the individual survey asked: "*On a scale of 1 (no contribution) to 5 (significant contribution) please rate the contribution of the following ZR features to your educational experience*". The results are shown in Fig. 31. Features that are specific to inter-team collaboration are marked in green. Among the purely collaborative features, 75% reported gaining from in-game collaboration (Collaboration Environment #1 in Section 4.1) and forums and challenges (Collaboration Environment #3). Less than 70% reported gaining positively from alliance-based collaboration (Collaboration Environment #2). It can be argued that the survey responses in Fig. 31 are heavily influenced by hindsight bias – since all evaluations are based on a post-tournament survey – and interference bias – too many collaboration variables were being evaluated at the same time. (People are likely to make errors in judging the individual impact of each factor.) Future editions of the ZR program can achieve more precise evaluations by having participants fill out a short questionnaire between each evaluative phase of the tournament. This will allow the factors that would produce bias to be better isolated. Too many questionnaires may also irk the participants, so a balance must be struck.

A more detailed analysis of 201 responses (only alliance participants among 246 total responses received) compared the relative significance of the ZR features listed in Fig. 31 in terms of their educational benefits to users. The relative preference of each individual for a specific ZR feature was calculated by subtracting two corresponding responses. This was repeated for all 201 responses and the histograms of differential preferences plotted in Fig. 32. The histograms were found to be normally distributed so calculating the mean differential preference (Fig. 33 as a color map) between every two ZR features were enough to specify the pattern of preferences.

There are few significant differences in relative preference among ZR features #4 to #7 i.e., the inter-team collaborative tools – marked green in Fig. 31 – and little relative preference between ZR feature #2 and #3 i.e.,

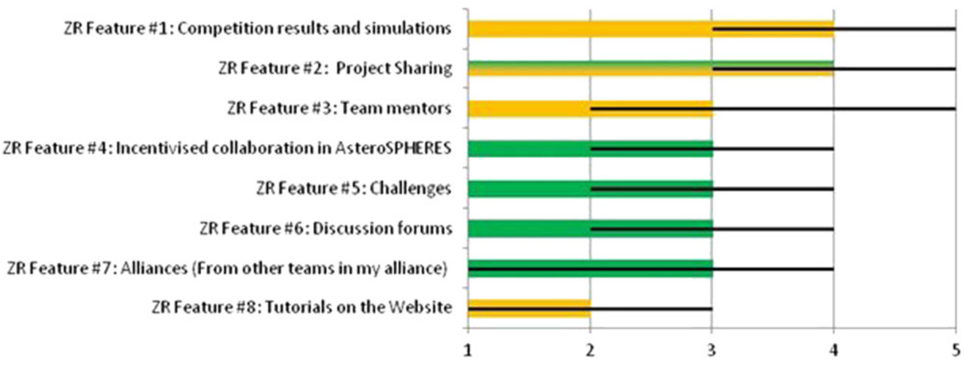

Fig. 31. Median of responses to:"On a scale of 1 (no contribution) to 5 (significant contribution) please rate the contribution of the following ZR features to your educational experience." Error bars indicate the inter-quartile range of responses. 'Green' indicates the inter-team collaborative tools. Project Sharing is marked half in green because it may be used to promote collaboration within a team or outside of a team, within an alliance. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

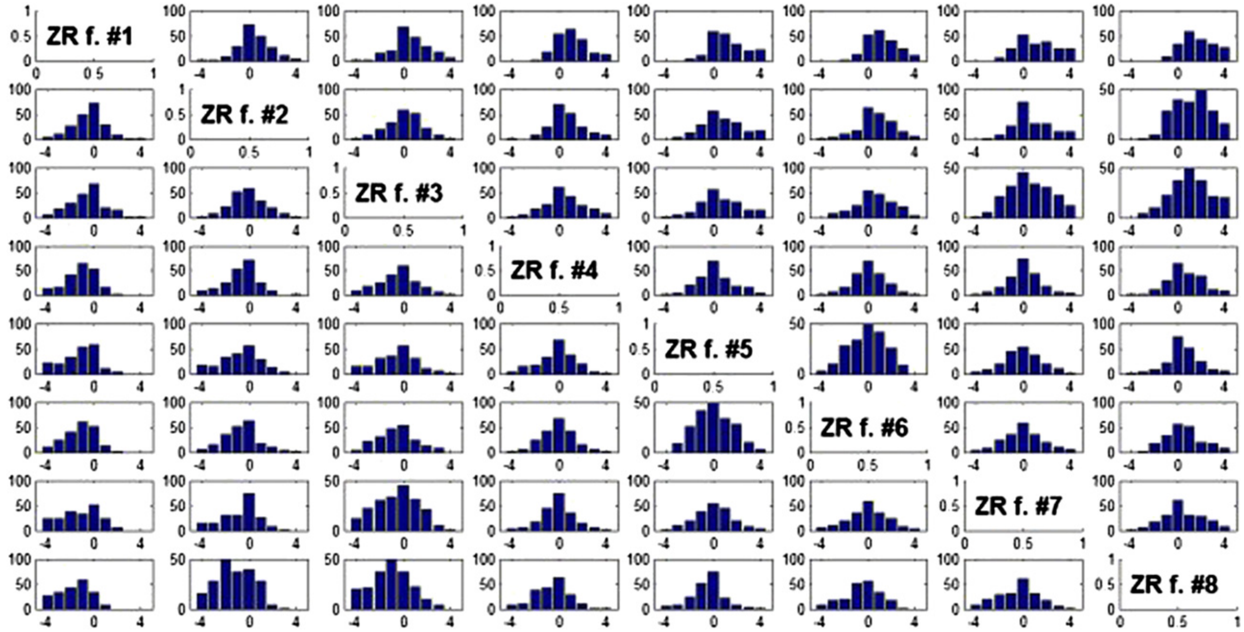

Fig. 32. Histograms of differences between pairs of preferences for the 8 ZR features in Fig. 31. The histogram in the Square *X*–*Y* represents the distribution of those differences for all subjects. For example, the histogram in the (*X*, *Y*)=(1,8) shows the distribution of the differences between the responses i.e., the preference for 1 over 8 ZR feature #1 and ZR feature #8. The range of preferences is −4 through +4.

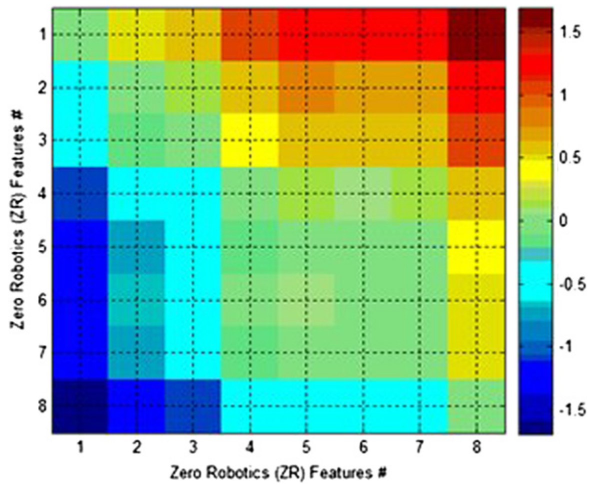

Fig. 33. Color map representing the average preference between the 8 ZR features in Fig. 31, taken two at a time. The average for each 8 × 8 block is obtained by finding the mean of the histogram distribution for that block from Fig. 32. For example, the top right-most corner of the color map indicating 1.5 in crimson is the average difference of response values to ZR feature #1 and ZR feature #8. Note that the color map is anti-symmetric about the main diagonal, because mean(*X*–*Y*)= —mean(*Y*–*X*). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

intra-team tools – instruction by mentors and project sharing. This inference was made using Fig. 33 which shows two squares of green that indicate near-zero relative preference between the rows and columns it represents in the color map. Intra-team tools however proved to be more beneficial to the students than the inter-team collaborative tools. This is noted from the yellowish patch in rows 2–3 (intra-team features) and columns 4–7 (inter-team collaborative features). The website tutorials (#8) were of lowest value and the competition results and simulations published on the website (#1) of highest. In those, teams could learn from their mistakes and from others' strategies.

The nonparametric Friedman test was conducted on the survey responses to rate the ZR features #4–#7 (inter-team collaborative tools). The similarity in their relative values as seen in Fig. 33 was found to be not significant. The subjects did not agree on the relative ranks of the tested tools. Results of the pair-wise Friedman test run (Table 3) on the responses to the 4 inter-team collaborative tools show high *p*-values for all the entries i.e., no statistically significant difference between the responses. On the other hand, the low *p*-value between ZR features #2 and #3 (Table 3) and the color map in Fig. 33 shows that the subjects agreed that the two intra-team collaborative features were equally important.

Participants who participated in both 2010 and 2011 tournaments were asked to rate the learning through 2011s collaboration and more than 60% reported positive results on a 5-point Likert scale as seen in Fig. 34. The 2010 web interface had no collaboration features apart from external discussion forum which logged 142 posts (the website had 144 users). This is a very low number compared to the 2011 web interface which logged 5150 posts (the website had 1689 users), even when the post to user ratio is considered. Fig. 34 also shows that, on average, students who had participated in 2010 found C programming easier in 2011, indicating the CS-STEM value delivered by the program.

Table 3
Multiple comparisons test table indicated the results of the nonparametric, pair-wise Friedman test conducted on the responses collected about the inter-team collaboration tools numbered ZR feature No.4 through ZR feature No.7 in Fig. 31.

Responses to feature No.	Responses to feature No.	Statistic	*p* Value
Pair wise comparison of inter-team collaboration features			
7	6	0.066	0.947
7	5	0.110	0.912
7	4	0.795	0.427
6	5	0.044	0.965
6	4	0.861	0.389
5	4	0.905	0.366
Pair wise comparison of intra-team collaboration features			
3	2	1.844	0.067

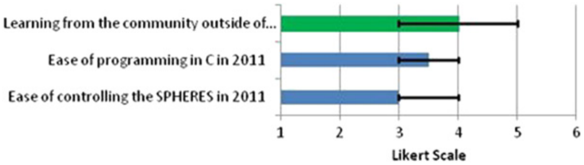

Fig. 34. Response of alumni from the 2010 tournament to "Please rate the following in the 2011 tournament with respect to your experiences in 2010" where 5=Significantly more, 4=A little more, 3=Felt the same, 2=A little less, 1=Significantly less. Error bars indicate the inter-quartile range of 'Green' indicates a question targeted to evaluate inter-team collaboration. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

6. Conclusions

This paper has attempted to assess the usefulness of *collaborative* games in space education based on existing theory that games are motivational learning tools and young adults are very fascinated by space. The hypothesis of usefulness has been attempted to be proven by the development of a hands-on educational robotics program followed by data collection over the last two years the form of the performance in the competitions, usage of the web interface, hardware operations on the ISS and feedback about the program to measure the utility of the tournaments. The analysis of data and the experience of running the program have taught us very valuable lessons in tournament design for efficient educational outreach within the ZR framework.

By allowing students to program real satellites using a high-fidelity simulator in an exciting video-game environment, the ZR program helps teach them math, physics, programming, strategy and communication i.e., 21st century skills, through engagement in real-world problems. ZR has successfully demonstrated tapping into the positive effects of games in the following ways:

- Each ZR game has a fictional but feasible story [15] to provide participants with an epic mission. The youth likes to save worlds and learn from heroes. A 'Star Wars' inspired droid (SPHERES) racing for revolutionary goals goes a long way far in inspiring them.
- The flash animation environment provides a sense of virtual worlds like a video game which allows programming to be fun and play and not just writing code.
- ZR provides the opportunity of an epic win [16] in a race that is literally out of this world. The incentive of ISS participation and astronaut interaction serves to motivate students all along. Also, since in the culminating event of ZR, all participants are invited to a common location to prepare for this 'epic win', the programming competition enters the real lives of people. Participants who had been corresponding and collaborating mostly over the internet can then meet each other and share the excitement.
- Games increase productivity by keeping up the sense of urgent optimism [16]. ZR allows racing among team members and scrimmaging against other teams. These online tools as well as closely spaced competitions, i.e., multiple short and long term goals, keep the pace of performance high through the tournament.
- ZR games aim to incentivize collaboration among opponents [19]. This is a valuable lesson for students because projects are increasingly becoming complex, and hardly any can be completed by an individual discipline, office or organization. Students work together as a team, outside of their teams in alliances and together with opponents to achieve game objectives. Collaboration in so many layers is expected to lead to exchange of knowledge and communal discovery. Students get a valuable primer that will help them in real world collaborative scenarios in the future.
- ZR games are strategy and mathematics intensive which encourages analytical thinking and pique the problem solving interest of many. It provides food for different skill sets within a team.
- Every ZR game has random variables and participants are expected to write players that can deal with the element of uncertainty. While the online tools give users the ability to tweak these variables, their random nature makes for unexpected and interesting twists in the competitions.
- The program is free of cost and completely web-based. It requires just mentor and student enthusiasm and very minimal resources, so it is easily accessible and quickly scalable.
- Each competition and challenge returns a large set of results. Consistent feedback of performance [19] allows teams to monitor their progress. Participants have the opportunity to review performances of all others and form alliances that are stronger than any of its individual parts, leading to more evolved players.

The program taps into real world spaceflight software algorithms, frames an interesting game around it and therefore tries to promote project-based learning, guided by mentors. Data collected over the last two years in. Overall, the program has shown success in less than 2 years of nation-wide operation. The program in 2010 has seen participation grow by 241% over 2010. Above 85% mentors and students have reported significantly positive improvement in CS-STEM and leadership skills, with moderate to strong correlation in opinions. ZR 2011 has fulfilled the motivations of the participants and the predicted retention rate is approximately 89%. Additionally, building on the existing theory that collaborative gaming is becoming a very powerful tool for learning and solving, we have introduced collaboration environments within ZR and attempted to assess the effect of these environments on the educational experience of the participants. Although the results obtained do not show conclusively positive results for all the collaboration environments in 2011, noticeable improvements due to collaboration have been observed. Overall performance of teams increased by 3.4 points on a 0-23 point scale. Participants attributed positive educational influence to all the collaborative features in ZR. Intra-team collaborative features were better received than inter-team features as indicated by their differential preferences, albeit at varying degrees of statistical significance. Most importantly, the feedback has shown us ways in which the collaboration implementation within ZR can be improved to deliver better quality education and we have a framework in place for measuring the effects on our objectives.

To conclude this paper, it must be stressed that there is a difference in the way performances are evaluated in achieving the dual objectives of crowdsourcing and STEM education i.e., the objectives of the ZR program overall [26]. In crowdsourcing, one cares only about the very best of solutions, i.e., for the rightmost tail of the histogram distribution of performances in any competition (Figs. 23 and 24). The purpose of sourcing solutions from dozens, hundreds or thousands of people is to identify the outliers that are most novel and high performing. For CS-STEM education, on the other hand, one cares to get maximum number of students involved and influenced i.e., shift the average of the histogram distribution for any competition toward the right or raise the average score (Figs. 25 and 27). The ZR program has proven that, in spite of aiming at a dual objective, it can successfully achieve its CS-STEM objectives.

Acknowledgements

This work was performed primarily under contract "NNH11CC26C: InSPIRE" with the Defense Advanced Research Projects Agency (DARPA) and also for contracts "NNH10CC27C: Supporting the SPHERES facility aboard the ISS for STEM education objectives" and "NNX10AJ90A: NASA Summer of Innovation—Zero Robotics" with the National Aeronautics and Space Administration (NASA). The author gratefully thanks the sponsors for their generous support that enabled this research. The subcontractors of the program are Aurora Flight Sciences and TopCoder. The SPHERES program is run by the NASA SPHERES team at Ames Research Center. The original concept for Zero Robotics was conceived in cooperation with astronaut, Gregory Chamitoff. The PI of the SPHERES program is Prof. David W. Miller and of the DARPA InSPIRE program is Prof. Jeffrey A. Hoffman. We would like to thank Alan Natapoff from the Man Vehicle Laboratory at MIT for his generous time and effort in helping with the statistical analysis presented.

References

[1] Rodger W. Bybee, What is STEM education? Sci. AAAS 329 (2010). 27-Aug-.
[2] C. Solomon, M. Minsky, B. Harvey, Introduction to LogoWorks. Byte Books, McGraw-Hill, New York, 1986.
[3] P. Gonzales, et al., Highlights From TIMSS 2007: Mathematics and Science Achievement of U.S. Fourth- and Eighth-Grade Students in an International Context," National Center for Education Statistics, Institute of Education Sciences, U.S. Department of Education, Washington, DC, 2008 NCES 2009–001 Revised.
[4] "OECD Report of Performance Ranks per Country." [Online]. Available: 〈http://geographic.org/country_ranks/educational_score_performance_country_ranks_2009_oecd.html〉.
[5] "CIA World Factbook." [Online]. Available: 〈https://www.cia.gov/library/publications/the-world-factbook/index.html〉.
[6] B. Trilling, C. Fadel, 21st Century Skills: Learning for Life in our Times, Jossey-Bass, San Francisco, CA, 2009.
[7] B. Trilling, From libraries to learning 'libratories'□: the new ABC's of 21st-century school libraries, Sch. Lib. Mon. 27 (1) (2010) 43–46.
[8] Cathy N. Davidson, Now You See It: How the Brain Science of Attention Will Transform the Way We Live, Work, and Learn, Viking Penguin Books, 2011.
[9] "NASA Strategic Plan 2011." 24-Apr-2012.
[10] Matthew Allner, et al., NASA's explorer school and spaceward bound programs: insights into two education programs designed to heighten public support for space science initiatives, Acta Astronaut. 66 (2010) 1280–1284.
[11] A. R. Johnson, et al., "International Space Station National Laboratory Concept Development Report," NASA Headquarters, NP 2007 03 459 HQ, Dec. 2006.
[12] Mark Craig, "Letter to NASA Headquarters: Massive Public Engagement in Space Exploration," 16-Nov-1998.
[13] National Academies Press, "America's Future in Space: Aligning the Civil Program with National Needs," National Research Council, Washington, DC, 2009.
[14] National Aeronautics and Space Administration, "Augustine Commission Report," JSC-Houston, Texas, Jul. 2009.
[15] Stuart L. Brown, Christopher C. Vaughan, Play: How it Shapes the Brain, Opens the Imagination, and Invigorates the Soul, Penguin Publications, 2010.
[16] Jane McGonigal, Reality Is Broken: Why Games Make Us Better and How They Can Change the World. 2011.
[17] Jesse Schell, The Art of Game Design: A Book of Lenses, Elsevier/Morgan Kaufmann, 2008.
[18] Edward Castronova, "A Test of the Law of Demand in a Virtual World: Exploring the Petri Dish Approach to Social Science," Jul-2008.
[19] Tom Chatfield, "7 ways games reward the brain," 2010.
[20] Piers Harding-Rolls, "Subscription MMOGs: Life beyond World of Warcraft," 30-Mar-2009.
[21] Nick Yee, "The Psychology of MMORPGs: Emotional Investment, Motivations, Relationship Formation, and Problematic Usage," in Avatars at Work and Play: Collaboration and Interaction in Shared Virtual Environments, vol. 34, London: Springer-Verlag, pp. 187–207.
[22] Jeffrey Kim, P.Allen Jonathan, Elan Lee, Alternate reality gaming, Commun. ACM 51 (2) (2008) 36–42.
[23] Helena Cole, Mark Griffiths, Who plays, how much, and why? Debunking the stereotypical gamer profile, J. Comput. Mediated Commun. 13 (4) (2008) 993–1018.
[24] Dmitri Williams, Nick Yee, Scott Caplan, Ching–I Teng, Personality differences between online game players and nonplayers in a student sample, Cyber Psychol. Behav. 11 (2) (2008) 232–234.
[25] Helena Cole, Mark Griffiths, Social Interactions in Massively Multiplayer Online Role—Playing Gamers,"CyberPsychology and Behavior, 10, 4, 583 575.
[26] S. Nag, Collaborative Competition for Crowdsourcing Spaceflight Software and STEM Education using SPHERES Zero Robotics, Dual

S.M., Massachusetts Institute of Technology, Cambridge, Massachusetts, U.S.A, 1965.
[27] "DARPA Call for Crowdsourcing Ideas for Cluster Flight Software," Federal Business Opportunities, 11-Feb-2010. [Online]. Available: ⟨https://www.fbo.gov/index?s=opportunity&mode=form&id=def9ae35c237359f0bc75dd730e8ed54&tab=core&_cview=0⟩. (accessed: 27-Apr-2012).
[28] S. Nag, I. Heffan, A. Saenz-Otero, M. Lydon, "SPHERES Zero Robotics software development: Lessons on crowdsourcing and collaborative competition," in Aerospace Conference, 2012, IEEE Xplore (10.1109/AERO.2012.6187452), Big Sky, Montana, U.S.A., 2012.
[29] Alvar Saenz-Otero, "Design Principles for the Development of Space Technology Maturation Laboratories Aboard the International Space Station," Ph.D., Massachusetts Institute of Technology, Cambridge, Massachusetts, U.S.A., 2005.
[30] J. Enright, M. Hilstad, A. Saenz-Otero, and D. Miller, "The SPHERES Guest Scientist Program: collaborative science on the ISS," in Aerospace Conference, 2004. Proceedings. 2004 IEEE, 2004, vol. 1, p. 46 Vol. 1.
[31] Swati Mohan, "Quantitative Selection and Design of Model Generation Architectures for On-Orbit Autonomous Assembly," Ph.D., Massachusetts Institute of Technology, Cambridge, Massachusetts, U.S.A., 2010.
[32] Alan Melchoir, et al., More than Robots: An Evaluation of the FIRST Robotics Competition Participant and Institutional Impacts, Heller School for Social Policy and Management, Brandeis University, Waltham, Massachusetts, U.S.A, 2005 Apr.
[33] A. Saenz-Otero, J. Katz, S. Mohan, D. W. Miller, and G. E. Chamitoff, "ZERO-Robotics: A student competition aboard the International Space Station," in Aerospace Conference, 2010 IEEE, 2010, pp. 1–11.
[34] MIT ZR Team, "Zero Robotics 2011 Game Manual," *Zero Robotics*. [Online]. Available: ⟨http://www.zerorobotics.org/web/zero-robotics/tournament-details?tournamentId=1⟩. (accessed: 28-Apr-2012).
[35] President's Council of Advisors on Science and Technology (PCAST), "Prepare and InspireL K-12 Education in Science, Technology, Engineering and Math (STEM) for Aamerica's Future," Executive Office of the President, Washington, DC, Sep. 2010.
[36] Federal Inventory of STEM Education Fast-track Action Committee on STEM Education National Science and Technology Council, "The Federal Science, Technology, Engineering and Mathematics (STEM) Education Portfolio," Washington, DC, Dec. 2011.
[37] D.T. Campbell, J.C. Stanley, Experimental and Quasi-Experimental Methods for Research, Johns Hopkins University: Houghton Mifflin Company, 1965.
[38] Earl Babbie, The Science of Social Research, 12th ed. Wadsworth Cengage Learning, 2010.