

2014 AASRI Conference on Sports Engineering and Computer Science (SECS 2014)

## An Integrated Computer Assisted Training System for the Baseball Defense Concepts and Cases

Chueh-Wei Chang<sup>a\*</sup>, Shih-Yang Lee<sup>b</sup>, Shih-Kuei Huang<sup>b</sup>, Chin-Long Yang<sup>b</sup>

<sup>a</sup>*Department of Computer Science and Information Engineering,  
National Taipei University of Technology, Taipei, Taiwan.*

<sup>b</sup>*Department of Physical Education, Chinese Culture University, Taichung, Taiwan.*

---

### Abstract

The successful execution of a baseball strategy in time is very important in the baseball game. The key point for this successful execution capability of a player does not only depend just on the daily routine practice, but also depend heavily on the correct concepts a player has. In this paper, we design and implement two animation assisted sport simulation subsystems, the Cutoff Play Training Subsystem and the Editable Fielding Strategy Training Subsystem, for general baseball fielding training. These two integrated training subsystems allow users to choose what happens once the baseball is put into play, and provides many different scenarios and cases that probably will occur in the baseball field. By using these animation simulation subsystems, players, especially those at the amateur junior level, can improve greatly on all aspects of fielding plays in a short period of time.

© 2014 The Authors. Published by Elsevier B. V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/3.0/>).

Peer-review under responsibility of Scientific Committee of American Applied Science Research Institute

**Keywords:** Baseball Fielding Concept Training; Animation Platform; Tactical Cases Simulation; Temporal Spatial Relationship;

---

---

\* Corresponding author. Tel.: +886-2-27712171 #4241; fax: +886-2-87732945.  
E-mail address: [cwchang@ntut.edu.tw](mailto:cwchang@ntut.edu.tw).

## 1. Introduction

A baseball game relies on teamwork, and consists of two basic parts: offense and defense. Defense plays an integral role in the success of a team. The team that makes the fewest number of mistakes in fielding will likely be the team that wins the game. In defensive plays, every player has a specific task that needs to be completed. Coaches need to instruct all the fielders involved in a play on defense strategies, i.e., how to properly execute a play quickly and instinctively. Fielders must comprehend their proper roles and the most efficient responses to various situations that can occur. Among the most widely instructed defense strategies are the “cover, relay and cutoff” methods designed for stopping the base runners from advancing bases [4]. These strategies require a great deal of practice and teamwork in order to consistently result in successful outcomes. There are many standard paradigms in the baseball textbooks for the general defensive cover, relay and cut-off play drills [3][6]. But, these standard paradigms are planar and static still images. These static graphs are unable to illustrate the movement and timing between fielders and runners.

In recent years, with the help of information technology and computer software, it has become easier to use animation simulation in athletic training [1][5] and competition management [7]. These technologies also can provide an excellent foundation for simulation based training. Therefore, we designed two economic, portable, reusable and flexible training tools, the Cutoff Play Training Subsystem and the Editable Fielding Strategy Training Subsystem, can help train players, especially amateur junior student players, to understand and remember each aspect of defense “cover, relay and cutoff play” situations and to study many different salient play cases. We followed a standard paradigm commonly used by the Taiwan national baseball team, and present a methodology to design and implement an animation based simulation subsystem that can help the general training of the “cover, relay and cutoff” play.

## 2. System design

### 2.1. Ball field object classes design

Before we started to design this simulation system, we needed to understand the actions that take place on a baseball field. Baseball is a team sport comprised of many actions such as throwing, catching, batting, and running. According to the baseball rules [2], a pitcher throws a baseball toward home plate, a batter attempts to hit the ball with a bat into the field of play, a batter who hits the ball into the field must begin running toward 1B and beyond, a fielder tries to catch the ball cleanly and throw to the proper location in order to stop the runner from advancing on the bases, especially to home plate. There are nine defensive players on the field, and at minimum one, up to a maximum of four offensive players on the field for a given play. We treat the baseball and each player as an independent “object” and each object has its own objective, such as catching a ball within their assigned defensive area, moving to cover a particular base, throwing the ball to the correct place, running to the next base, or just moving to a designated location, etc.

To design this system platform, we used a tree-structured scene description to define the spatial and temporal position of these objects and their movement in a given simulation. The system's compositor uses the scene description information, together with each objects data, to determine the final outcome. We use the 45 degrees view projection to create our view of the baseball field, and use Adobe Flash and Action Script language to create a Windows user interface, and generate a series of animations. A full simulation consists of many animations of various objects which perform actions in adherence to the cutoff play and the base running strategy rules. For example, a fielder's action is displayed by a fielder moving (running) animation, a fielder catching animation and a fielder throwing animation accompanied with ball movement animation.

## 2.2. Systems operation work flows

These two subsystems' work flows are shown in Fig. 1 (a) and (b). For the Cutoff Play Training Subsystem, at the beginning of each play, the nine defensive players are in their normal field positions. Also, we put the runner(s) of the batting team on the base(s) according to the user's selected parameters. Before we start the simulation, several factors must be clear: who will handle the ball first after the ball is hit, who will play the cutoff man and in which location, who will cover which base, how many bases the batter-runner and runner(s) can advance, and who will backup for an error and in which location. The work flow of the Cutoff Play Training Subsystem is consisted by five steps as followings.

- **Setup and Initial Objects** – This step will put all objects, including the ball field background bitmap, defensive and offensive players, and the baseball, onto the simulation scene. Each object has its own associated animation (running, batting, pitching, etc.) and pre-defined starting location according to standard paradigms.
- **Load Fielding Zone Map** – In order to define the area each of the nine fielders are responsible for, we designed a color fielding zone bitmap. The size of the fielding zone map is exactly the same as the ball field background image. We use the color of the ball landing location in the corresponding field zone map to assign which fielder will catch the ball without any ambiguity between fielders.
- **Ball Field Situation Setting** – Before we start to simulate a play, we need to set up four important factors: (a) the ball hit status, (b) the runner(s) on base(s), (c) the number of outs, (d) hit/out batting result. The ball hit status means the movement of the ball after it is hit. It could be a fly ball, line drive, or ground ball. The runner(s) on base(s) have eight different combinations from bases empty to bases loaded. The number of outs is from 0 to 2. There are five different options for the hit/out batting result we can simulate: single, double, triple, fly out, and bunt. Once these four parameters have been selected, the ball landing location also needs to be specified. To simplify the case selection, we provide 18 commonly used training examples.
- **Scenario Simulation** – After the scenario situation has been selected, the system can, just like players in a game, make decisions as to their subsequent actions from the time the pitcher delivers the ball until the play is completed.
- **Message Output** – After completing the simulation, the system will display the fielders' movement paths, ball hit and ball passed trajectories, and corresponding coaching advice that a fielder must know. These output messages provide sufficient information and details about what each player should do in such a situation.

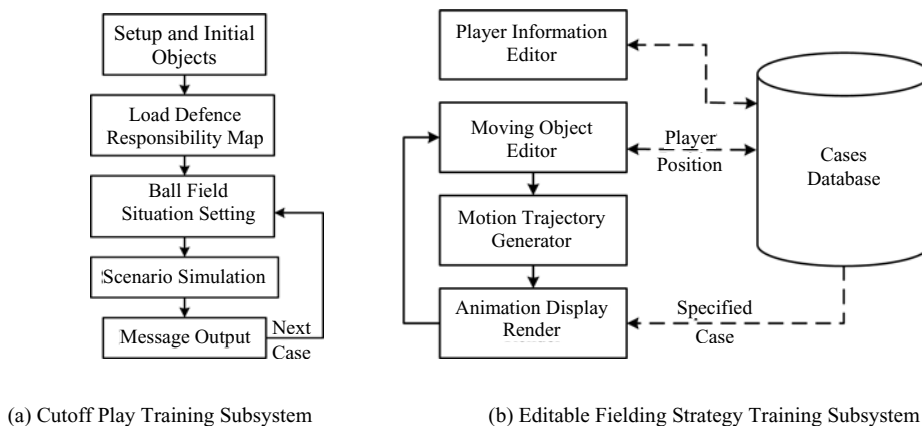


Fig.1. System operation block diagram

For the Editable Fielding Strategy Training Subsystem, it is designed as an editable version of training animation tool with a database for coaches to create their own tactical paradigms and store special cases. The cases of baseball strategies, no matter currently used or selected from the past, can be easily created, collected, reedited and stored in this system. It can also provide the moving trajectories of the ball and players in the ball field, and can display the relationship between spatial position and time interval by a computer assisted animation way. This editable version can be used as recording and replay the movement of each object when the user changes an object's position according to time. The work flow of the Editable Fielding Strategy Training Subsystem can be divided into four steps as followings.

- Player Information Editor – In order to show a real world case, user can give each player object their related information, such as the name and number, and store into the cases database.
- Moving Object Editor – User then needs to change the position of each player object according to the change of the ball object's moving direction step by step. That is, each time a change of the ball direction, user needs to create a new key point. Every key points and related positions of the objects will be stored into the cases database also.
- Motion Trajectory Generator – When this subsystem starts to simulate the case following a timer, each object will recalculate their positions by following the key points and the time intervals specified by the Moving Object Editor.
- Animation Display Render – Once the timings and positions of each object have been decided, this subsystem calls the Flash Player API to generate a series of animation.

### 2.3. Decisions making

However, there are many diverse and complex scenarios in a baseball game. Our Decision Making Method only focuses on the most common cases and ensures there were no mistakes made by the fielders. Our Decision Making Method for each player is contingent upon where the ball lands and its movement. For example, on a single to left field with no one on base, the left fielder will move to field the ball and throw to the cutoff man (shortstop), the shortstop will move to align himself with the left fielder and second base to prepare for a potential cutoff throw, and the center fielder will move to back up the left fielder in case the ball eludes or deflects off his body.

Once the players have taken their appropriate positions on the field, we can divide the time of the play into several intervals and use the Critical Time Point method to evaluate and adjust, if necessary in accordance with the national team standard paradigm. In each critical time point interval, we need to determine which fielder will catch the ball and where the fielder will throw to next. We also need to decide what kind of action the other players will do in the situation. Therefore, we combine and apply the standard paradigms into our internal system structure.

- Critical Time Points Selection – In each play, pitching, batting, catching and running all have their own particular actions and methods. We divide each play into four time intervals, as shown in Figure 5, for each time the ball changes direction. The time interval  $t_0$  to  $t_1$  represents the time it takes for the ball to travel from the pitcher to home plate. The time interval  $t_1$  to  $t_2$  represents the time it takes once the ball is hit by the batter until it is fielded. The time interval  $t_2$  to  $t_3$  represents the action of the receiving fielder till he throws to the cutoff fielder. The time interval  $t_3$  to  $t_4$  represents the interval for passing the ball from the cutoff man to a specific baseman.
- Fielding Player Selection – After the pitcher delivers and the batter hits the ball, we must determine which fielder will handle the ball first. By using the color-coordinated fielding zone function the correct fielder is easily identified. Once identified this fielder needs to move to where the ball lands and field the ball. The

remaining eight fielders move to their specified places to cover bases, cutoff and relay, or backup in the event the ball gets passed the determined fielder.

- Cutoff Player Selection – Next, we need to decide who will be the cutoff man according to the standard paradigm. Sometimes, we need to select more than one cutoff man if there is more than one runner, or another complicating factor. The cutoff man needs to move into a proper position, which is most likely the midpoint between two fielders, the one fielding and throwing ball and the one covering the base.
- Base Runner Moving Method – At most, there are four base runners running at the same time. For each runner, how many bases he can advance will heavily depend on where ball lands, the number of outs, and whether its' a ground ball or a fly ball. For example, if the ball is hit deep into right field and a runner is on third base, regardless of whether the ball is caught the runner on third base has a chance to cross home plate and score.

### 3. Implementations

We use an object-oriented programming language Action Script associated with Flash Player API to implement this animation based system. The Flash Player API is made up of classes that represent and provide access to object animation function. Fig. 2 and Fig. 3 are our two subsystem user interfaces.



Fig.2. Cutoff Play Training Subsystem User Interface



A user can start the Cutoff Training simulation by selecting a location where the ball lands with his mouse in the ball field area and pushing the “Start Demo” button on the bottom right corner of the window. The commonly used 18 typical training examples that we provide, such as all bases are empty for single hit to left field, are put in the “Options” area. Several radio buttons are designed for toggling between showing or hiding the trajectory lines of the ball and the fielders. There is also a description for each player in the upper right “Description” area, such as, the shortstop lines up a throw to 2B and possibly cuts and relays that throw to 2B, the second baseman covers 2B, the right fielder moves to backup 2B in line with the throw, the third baseman covers 3B, the catcher trails the runner to 1B and backs up 1B, the first baseman makes sure the runner touches the base and then covers 1B. The user can repeat each scenario and learn where he should go, what he should do, and how to collaborate with his teammates in each such situation.

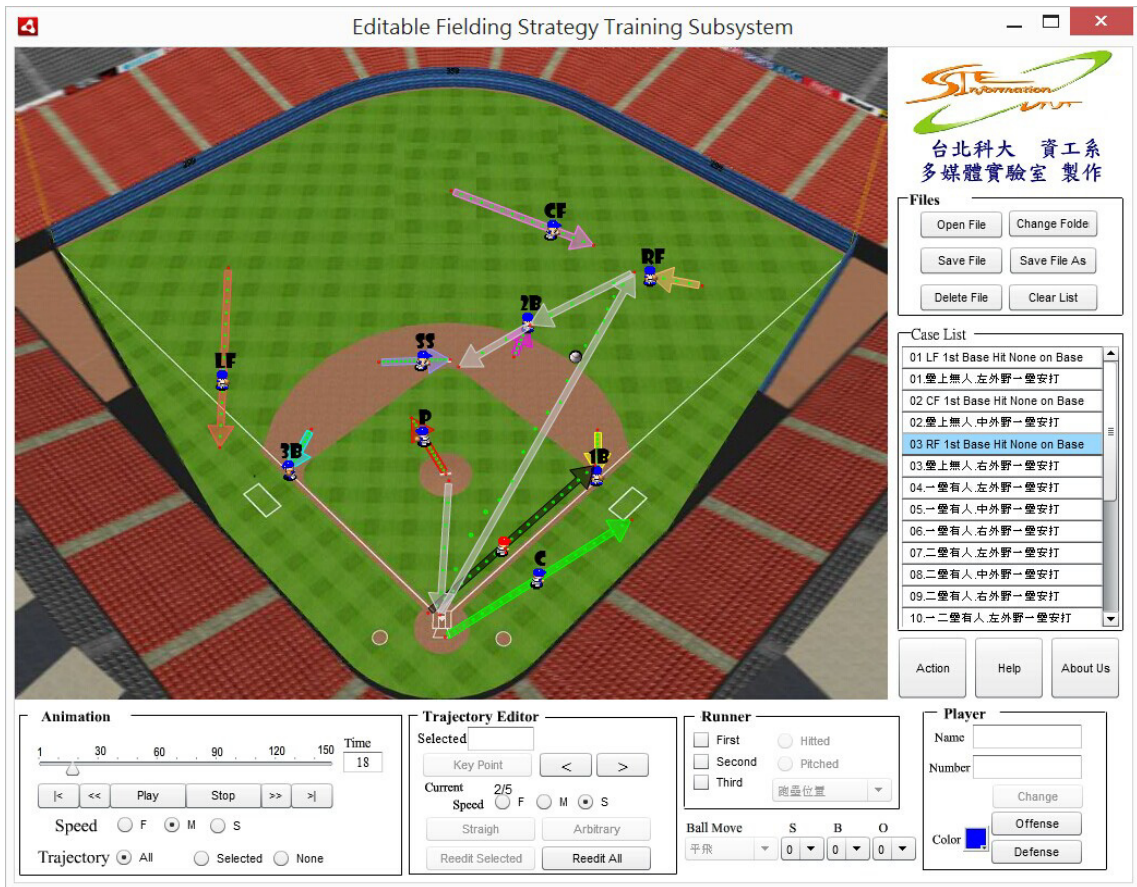


Fig.3. Editable Fielding Strategy Training Subsystem User Interface

#### 4. Conclusions

These two simulation subsystem are economic, portable, reusable and flexible training tools that can help baseball players have a better understanding and retain the concepts of each aspect of various defensive situations. By using these subsystems, coaches and players can get a better training performance and increase

communication efficiency. The strategy cases, after the editing processes, can be stored as digital files permanently, reviewed repetitively, shared with others and reedited easily. Following our hierarchy design method, this animation platform can be easily applied to many kinds of other sports.

## References

- [1] Chang CW, Lin HW and Chang CK. The construction and applications of a baseball strategy database with animation playback. *Sports Coaching Science* 2005; 5:147-158.
- [2] Commissioner of Baseball. Official baseball rules. Major League Baseball Office of the Commissioner 2011.
- [3] Johnson M, Leggett J and McMahon P. Baseball skills & drills. Champaign, IL: Human Kinetics; 2001.
- [4] Kindall J and Winkin J. The baseball coaching bible. Champaign, IL: Human Kinetics; 2000.
- [5] Leser R, Uhlig J and Uhlig M. Development of an application for learning and teaching soccer tactics. *International Journal of Computer Science in Sport* 2009; 8(2):19-31.
- [6] Stallings J and Bennett B. Baseball strategies. Champaign, IL: Human Kinetics; 2003.
- [7] Vincent J, Stergiou P and Katz L. The role of databases in sport science: Current practice and future potential. *International Journal of Computer Science in Sports* 2009; 8(2):50-66.