Final Lab Write-Up
by Edward Ekstrom and Spencer Weight

Declaration of Time Spent

Edward Ekstrom

Edward completed these items:

- Implemented our Kalman filter to work in conjunction with other agent types.
- Wrote these sections of the report
    - Ability to Target and Hit Enemy Tanks
    - Server Settings Adjustments
    - Special Strategies Employed

Edward worked 5 hours on the project

Spencer Weight

Spencer completed these items:

- Programmed the final_agent that assigned agent types to each tank
- Wrote these sections of the report
    - Flag Retrieval
    - Coping With World Noise
    - Quality of Implementation

Spencer worked 9 hours on the project

Teams we competed with

- Ryan Palmer and Craig Jacobson
- Dean Lebaron and Chris McNeil
- Trevor Buckner and Sam Gubler
- Brooke Maxwell and Ian Bazinger
- Trevor Dixon and Brian Smith

Flag Retrieval

Us

Our agent's method of flag retrieval is a simple one. First each tank on the team is assigned an agent (Kalman or Potential Field). If the tank is given a Potential Field Agent then it will begin the match by placing an attractive potential field on the nearest flag. This attractive potential field will draw the tank towards it until it picks up the flag. Once the flag is picked up, the tank switches gears and places an attractive potential field on its respective home base. The tank heads for home until it drops the flag off. Once the flag is dropped off, the tank then reverts back to its initial goal of heading for the nearest flag.

Them

Each team that we played against employed the same tactic of heading for the nearest flag. One team had their tanks making decisions based on utility. The tanks would head towards the nearest flag, but if an enemy was within range, their utility calculation would have their tank shoot at the enemy instead of pick up the flag. The most commonly used way of directing tanks toward the nearest flag was to use an attractive potential field. This is because it was less processing time than having to calculate the distance every single time, and it was also our first project that was consequently the easiest.

Ryan Palmer and Craig Jacobson's agent was good at flag retrieval. In our second game against them we had a huge lead when they killed our last flag seeking tank. We still had all three defensive Kalman tanks but their one remaining flag retrieving tank was not going after our flag but going to the blue flag that was north but in the middle of both of us. Since they were going back and forth between their base and the north blue base, they had the cover of the Ls and the middle pillar to shield them from our three Kalman tanks' accurate shots. Finally, as they were about to pass us point wise, one of our Kalman tanks threaded a bullet between an L and the center column, leading their last flag seeking tank perfectly and killing it in its tracks! It was awesome and this allowed us to hold our lead as time expired! This was probably the most exciting game we had.

Coping With World Noise

Us

The method we used to cope with world noise was to not do anything at all. We exploited the fact that if an agent has too much to process, then it cannot command the tank quickly enough, so the tank follows an oscillating path that allows it to circumvent obstacles on many of the simpler maps (four

L's, rotated box word, etc). Initially we tried using a tank that would act as a scout, but we decided that the scout tank would be more useful as an offensive tank rather than a passive one. Since we didn't have a scout tank, we also toyed with the idea of detecting whenever a tank got stuck and then changing its path accordingly to move it out from behind the obstacle. This didn't pan out and we ended up cutting out this idea from our code. We were able to justify this because we observed that without this code, the tanks were rarely getting stuck long enough to get shot.

Them

The methods that the other teams had for dealing with world noise were similar to ours, but there is one team that made a considerable effort to map the word. This team we played against had the same idea of creating a scout tank that would explore the world and identify obstacles. They sent out a scout tank that would search for "high density" areas, and then it would identify the corners of the obstacles. Once the corners were known, tangential fields were assigned to those spots to push any passing tanks around them. Their method wasn't perfect because their tanks still managed to get stuck or caught in infinite loops.

Quality of Implementation

Us

We felt that the quality of our implementation was good because we were able to win all of our matches except for the four team battle where we took second place. The implementation of our final agent was pieced together like Frankenstein's monster. Instead of rewriting any parts of code, we decided that all our code was fine just the way it was, and so we wrote an additional class that had a main function where it kept a list of agents assigned to each tank. We hard coded the list length to 10 with the intention of being able to figure out how to dynamically change it later. We also hardcoded 10 tank assignments; 3 tanks would be Kalman agents to stand as turrets and defend the base, and 7 tanks would be attackers that would head to the nearest flag.

The Kalman agent worked extremely well. It was able to command the Kalman tanks to shoot down incoming tanks with precise accuracy. One thing that we wish could have been done would have been to have the tanks patrol around the flag and still maintain their accuracy.

The Potential Field agent worked well, but it was not smart. All the commands that it sent to the Potential Field tanks told them to shoot continuously. This was a problem if tanks were allowed to shoot their own teammates. The initial blast of shots when the tanks began moving would take out half our team. Since the lab spec said to have the "--friendly-fire" flag set, this was not a problem. In fact, we never lost to a team if the "--friendly-fire" flag was on.

Once our Potential Field tanks got moving they had a chance to get stuck if they ran into a wall that was perpendicular to their desired path of travel. Another problem that our potential field tanks had was that they could not get around any of the stationary Kalman tanks if they ran into them. This, on a few experiments against ourselves, prevented our tank from returning successfully with a flag.

Them

Trevor Dixon and Brian Smith's agent beat us with the setting that stopped friendly fire shots from killing our own tanks turned off. This was because our initial shot blast would destroy half our team. Their tanks did not begin shooting until they saw an enemy, and so they did not have this problem.

Trevor Buckner and Sam Gubler's implementation did a good job of having their tanks move in formation. This worked well for them because if a single tank shot their group, then the formation would turn and take out the single tank and then continue on.

Dean Lebaron and Chris McNeil let us have a peek at the first section of the paper that they wrote on the implementation of their tanks. Upon reading their descriptions of the implementations of the various types of tank agents, it seems that they put a lot of effort in diversifying the methods of meeting the various problems that arose on the battlefield. They had a few scout tanks to find obstacles, they had a few tanks do some all out attacks, and they had a few tanks defend the base using the Kalman filter. If implementation quality is determined by being able to handle a variety of problems, then this team's implementation is the highest quality.

Ability to Target and Hit Enemy Tanks

Us

We had a great ability to target and hit enemy tanks. Our three defensive Kalman tanks would not be as accurate when there were tons of tanks left on the field because so many commands were being sent to the server. Though, sense there were so many tanks for our Kalman tanks to aim at, they would still hit and destroy a lot of enemy tanks. Then, as the number of tanks decreased, our Kalman agent became more accurate and would have the precision necessary to hit those last few enemy tanks left on the field.

Them

Ryan Palmer and Craig Jacobson's implementation was pretty good at targeting and hitting enemy tanks. The only problem was that if they had the flag and were on their way to return it to their base and they came close to an enemy tank, they would stop returning the flag and start targeting the enemy tank. This made it so they were hardly ever able to return enemy flags for points.

Dean Lebaron and Chris McNeil's agent did not seem to be able to target and hit our tanks very consistently. It seemed like they were moving too much to be able to aim properly or something. They only took out a few of our tanks before we were able to wipe out their whole fleet and win the game by a landslide.

Trevor Buckner and Sam Gubler's agent was very good at targeting and hitting enemy tanks. Our implementation and theirs were very similar in that they would leave tanks behind to defend and sent the rest to attack. The difference was that they left four tanks back instead of three like us. At the beginning of our first game against them it seemed that our one extra flag seeker gave us a distinct advantage. We took the lead quickly capturing their flag about twice as often as they captured ours. As time went on, we killed all but one of their flag seeking tanks and they killed all of our flag seeking tanks. This allowed them to mount a comeback but when they were about to return the flag to pass us

point wise, their tank got caught in a tangential field infinite loop and time expired with us still ahead! It was a great match.

Our second game against Trevor Buckner and Sam Gubler we played on large four l's map. This time it was not as close. We took out all of their flag seeking tanks and defensive tanks and won by a landslide.

In the mass battle four player game, we were not able to see how well anyones targeting Kalman tanks really were because there was such a mass attack at the beginning that almost all defensive tanks were shot early on.

Server Settings Adjustments

Us

The server setting adjustments we made throughout our experiment battles against the other teams were trying it without the "--friendly-fire" flag, and trying it with different maps. Without the "--friendly-fire" flag turned on, our agent was horrible. We would shoot a shot from each tank as the game began, killing over half of our tanks within the first second.

On the small four Ls world, we beat everyone pretty soundly. We tried starting on adjacent bases and bases across from each other, but that never made any difference. We also tried playing a few teams on the normal four Ls world since their agents were optimized for that map. Sometimes it made it a closer battle, but we still won each game.

Them

Trevor Dixon and Brian Smith's agent was really good at playing without the "--friendly-fire" flag. Since they would not shoot an initial shot at the beginning of the game none of their tanks would die. They would also only shoot when one of our tanks was in whatever path they were traveling so this made it so they would never shoot their own tanks. Hence, they would beat us easily without the "--friendly-fire" flag turned on.

The other teams we played all would do the same thing as us and shoot an initial shot right at the beginning of the game, so the "--friendly-fire" flag would not make a difference against any of them.

We already commented above on the differences between the different maps we played against people on, but it never made that much of a difference which map it was for any of the other teams we played against.

Special Strategies Employed

Us

The special strategy we employed was to not map out the terrain of the map because this would overload the server and make our Kalman tanks less accurate. Since we could already know where all of the other flags were without mapping the terrain, we thought it was a waist to have tanks wondering around wasting commands to the server with grid mapping stuff. I think this was the key to our success against the other teams.

Them

Ryan Palmer and Craig Jacobson's special strategy was to have all of their tanks be a hybrid of flag seeking tanks and targeting tanks. This allowed them to be very versatile but it also made their implementation more complex, allowing us to get more flags than them since their tanks would get distracted by enemies when they had a flag.

Dean Lebaron and Chris McNeil's special strategy was to have three different classes of tanks (attackers, defenders, and scouts) but a tank's role would change over time. I really liked this idea and I think if they had had more time to implement it, their agent would have been very powerful. The problem they ran into was similar to Ryan and Craig's problem. If one of their tanks was an attacker, had a flag, and was returning it to their base, if it saw an enemy tank, it would immediately turn into a Kalman tank and stop and target the enemy.

Trevor Buckner and Sam Gubler's special strategy was to only use the Kalman filter to determine if an enemy was in a tanks line of site. If it was, they would fire a shot, otherwise they would never shoot and just go after the enemy flag. This is a great idea for a game where the server is not running with the "--friendly-fire" flag turned on because they would never shoot their own tanks, but otherwise it is not very good because they would kill less enemy tanks purely because they fired infrequently.