A5 - Optimization

Group 4: Taichi Aritomo, Kian Jackson, Justin Wu, Michael Parrilla CS428, Rutgers University

PART I - Optimization of Pillars

Objective Function

Assuming that the simulation has no crashes, the objective function can be written as

$$J(\theta) = -\left(\sum_{r \in R} \frac{\theta}{t_{AVG}(r)}\right) / |R|,$$

where θ is the number of agents who completed their task, R is the set of results, and $t_{AVG}(r)$ is the average time of the agents in result r, as defined by:

$$t_{AVG}(r) = \frac{\sum_{(x,y) \in r} t(x,y)}{|r|}$$

Each result r is represented as a set of pairs (x, y) for each agent, where x is the time elapsed (if available) of the agent, and y is a binary integer indicating if the agent completed. If the agent did not complete, then the time elapsed is taken from the simulation settings. This distinction is made with the function:

$$t(x,y) = \begin{cases} x, & y = 1 \\ (\text{NUMFRAMES} \,/\, \text{FPS}) \cdot \text{U}, & y = 0 \end{cases},$$

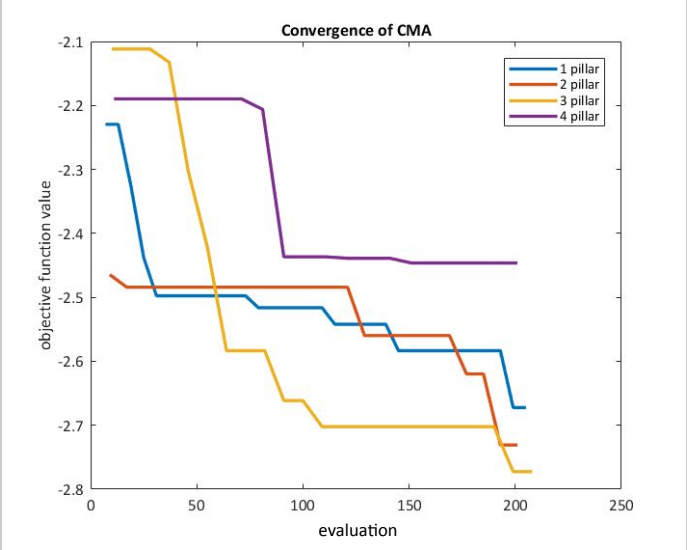
where NUMFRAMES, FPS, and U are constants provided in the simulation settings.

Videos

1 pillar, initial 1790 frames	2 pillar, initial 1945 frames	3 pillar, initial 3000 (MAX) frames	4 pillar, initial 3000 (MAX) frames
1 pillar, optimized 1602 frames	2 pillar, optimized 1462 frames	3 pillar, optimized 1374 frames	4 pillar, optimized 1714 frames

Analysis

The optimized parameters are considerably better than the initial parameters. For all of the test cases, the initial pillars do little to diverge the two directions of agents, and with more pillars walls are formed that block or even trap the agents.



For *1 pillar*, the initial setup did not affect most of the agents and it was like there was no pillar at all. When optimized, the one pillar in the middle worked to split the two opposing crowds, so that the agents moving leftward generally went above the pillar and the agents moving rightward generally went below the pillar.

For *2 pillar*, the initial setup again did not help to split the two crowds. Instead, it acted as more of an obstacle by creating extra traffic in the middle of the hallway. When optimized, the 2 pillars were centered and slightly offset from each other, directing the left-moving agents upwards and the right-moving agents downwards.

For *3 pillar*, the pillars initially create a wall in the middle of the hallway, even trapping one of the right-moving agents in its curved shape and preventing the agent from finishing. The optimized pillars work well to split the two directions again, but this time the pillars are spread apart more than they were in *2 pillar optimized*.

For *4 pillar*, the pillars initially create a wall, bigger than in *3 pillar*, and this wall traps four agents instead of just one. Once optimized, the pillars force the left-moving agents upward and the right-moving agents downward. However, it seems that CMA had a difficult time making *4 pillar* move the agents as efficiently as the other cases with fewer pillars. As the plot shows, the objective function value for *4 pillar optimized* was at least 0.3 greater than that of *1 pillar optimized*, *2 pillar optimized*, and *3 pillar optimized*. It is possible that, by increasing the maxFrames and allowing CMA to continue for a longer time, *4 pillar* may be optimized to a level comparable to the others.

PART II - Optimization of Social Forces AI

Notes

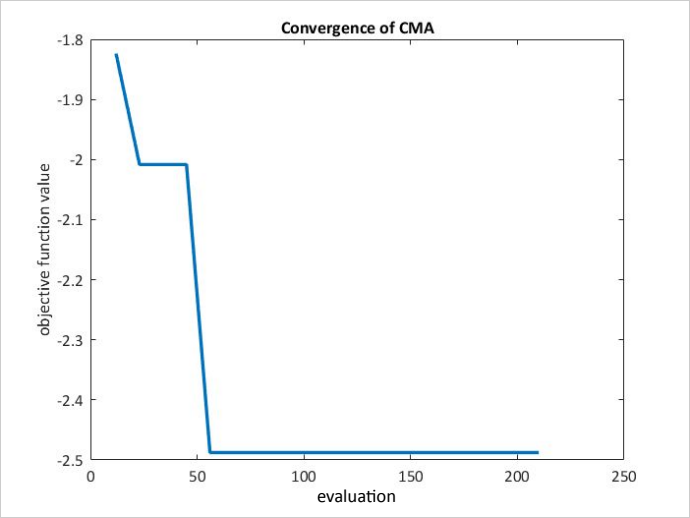
Upon running the CMA optimization for Social Forces AI with the initial parameters provided in `sf-param-config.xml` and maxFrames set to the default 3000, we saw that CMA terminated after the first iteration of simulations (12 evaluations) resulted in no significant improvement. This was indicated by the command-line report: `Reason for termination: tolfun`. Our guess is that the provided parameters in `sf-param-config.xml` were already close to optimal (at least locally) as far as the simulation could figure in 3000 frames.

Seeing that a significant number of simulations had been cut short in 3000 frames, we increased the maxFrames configuration to 5000 and ran the optimization again. This time, optimization actually led to incremental improvements in the objective function.

Videos

SF initial | SF optimized

Analysis



In *SF initial*, agents tend to collide violently, suggesting a low proximity force (collision avoidance) and high repulsion force (collision recoil) between agents. One can see that, when an agent gets pushed back by an opposing one, the push propagates to agents behind him, setting back the progression of a lot of agents in the hallway. Watching this, one might think of all the belligerent 16-year-olds moshing at a Tyler, the Creator show.

SF optimized depicts a gentler scene. It seems from the video that the proximity force is higher, and in the uncommon cases when two agents do touch, the repulsion force is not as strong. This is confirmed in a comparison of the initial and final parameters (Table 1). The multiplier for repulsion force between agents, *sf_agent_body_force* is reduced to a third of its original value, while the multiplier for proximity force between agents, *sf_agent_a* is more than doubled. Thus, the agents try harder to avoid colliding with each other, but the recoil is much softer when they do collide. Another parameter, *sf_personal_space_threshold* is decreased, suggesting the intuition that movement in a crowded space is more efficient when standards for personal space are lowered. Similarly, movement seems to be more efficient when agents are willing to get closer to walls, as suggested by the decrease in the wall

proximity force multiplier, *sf_wall_a*. From the plot, it's apparent that a significant improvement was seen between evaluations 45 and 56, and if the data could show the parameters of the best_ever values after each iteration, it might reveal what parameter changes are the "low hanging fruit" in optimizing social forces in this crowded hallway scenario.

Table 1: Comparison of Parameters in Social Forces

Parameter	**Initial Value**	**Δ**	**Optimized Value**
sf_personal_space_threshold	0.3	↘	0.1
sf_agent_body_force	1500	↘	500
sf_agent_a	25.0	↗	52.4126844
sf_wall_a	25.0	↘	6.21359110