Project 3: Optimization & Planning

Project Due: Sat., Nov 11

Overview. One of the most important modern trends in animation is the rise of optimization-based techniques. These approaches try to control some aspect of the animation by optimizing a "loss function" or "energy function". These functions are typically based on underlying physics (optimization-based physical simulation), matching recorded data of the desired system appearance (data-driven animation), or achieving a specific goal set by the animator (reinforcement learning). In HW3, we examined the optimization of an objective function. Here, we will explore planning-based techniques for creating intelligent agent motion.

You may work with a partner, just one project turn-in needed per pair. <u>You cannot repeat your partner from any of the previous projects or HW3.</u>

Part 1: Inverse Kinematics Scenario [up to 90 points]
You will need to create a simulation where an animated character uses inverse kinematics to move, grab, or reach in an environment. The basic requirements involve using a single IK solver to animate a very simple 2D skeleton. For additional credit, you should implement a 3D scenario, plan for more complex skeletons, add obstacle avoidance, and user interactions.
Required components are indicated with a star (*).

Single-arm IK (at least 2D)* (up to 20 points).
Produce an animation with a skeleton that has at least one arm and at least four joints. The end effector of the arm should be able to reach and touch an object. However, the root of the skeleton should remain stationary throughout the reach animation. Make sure to demonstrate at least one successful animation and one animation that shows what happens when the object is out of reach.

Multi-arm IK (at least 2D) (up to 20 points).
Produce an animation where a skeleton, of at least two arms, with at least three joints each, and at least two end effectors, reaches to touch an object. The skeleton must have a central root between the two arms that does not move during the reach animation, and one or more additional shared roots between the arms and the root (that can move for either arm during reaching animations). Both end effectors must be able to move and touch an object. You must show at least one successful animation for each arm, and one animation of what happens when the object is out of reach both arms. *[This will also count as single-arm IK meaning multi-arm IK totals up to 40 points.]*

Joint limits* (up to 20 points).
Add angle and/or rotational speed limits to some or all of the skeleton joints. Include in your report a video showing the difference in motion with and without

these limits. To earn full points, make sure to select limits that enhance the natural appearance of the resulting animations.

IK Obstacles (up to 20 points).
Add multiple obstacles to your environment. Detect and prevent any collisions between the skeleton and the obstacles while still allowing the skeleton's end effector to reach its goal.

User Interaction (up to 10 points).
The points for UI come from allowing the user to interact directly with the simulation itself (not from controlling the camera). To receive full points, the user should have a clear, smooth, and natural way to interact with the animation. Discrete interactions, such as toggling some behavior on/off, will only receive a couple of points. Look for continuous interaction, such as allowing the user to move an obstacle with the mouse or control where the IK system is trying to reach.

[You can get these points on either IK or a challenge simulation]

Moving IK (at least 2D) (up to 10 points).
Allow the skeleton to move its root in a natural looking fashion that interacts smoothly with the IK-based animation.

Re-rooting IK (at least 2D) (up to 20 points).
Allow the skeleton to walk, climb, or inch its way forward by changing which node is rooted to the ground. For example, walking could be done by rooting the left foot, swinging the right foot via IK to reach a point about a meter forward, then treating the right foot as the root of the skeleton and using IK to plan the left foot. Note: do not violate any of the length constraints of the skeleton. *[This will also count as moving IK meaning re-rooting IK totals up to 30 points.]*

Motion Planning (up to 20 points).
Use a PRM, RRT, or any other motion planning technique to enable your skeleton to plan a path through the environment. The path planning should either ensure that the root of the skeleton avoids obstacles or plan an optimal series of target points for the feet/hands (or plan both the root and feet/hands). To complete this requirement, the motion planner should solely plan the positions of the root and grip, while the IK solver must control all other joints.

3D Simulation (up to 10 points).
Simulate and render your IK system in a 3D environment. For full credit, the 3D nature of the motion needs to be clear in the resulting animation (e.g., through the use of dynamic cameras, 3D lighting, etc.).

3D Rendering & Camera (up to 10 points).
Render at least one of your simulations in 3D (only the rendering needs to be 3D, the simulation motion can be 2D). For full points, the camera should be easy to use naturally, the models easy to see, and the scene well-lit with a clear sense of depth.

Texturing your models and using multiple light sources are good ways to achieve this highest level of visual quality.
[You can get these points on either IK or a challenge simulation, a 3D rendering of a simulation with 3D motion will combine two features for 20 points]

Skinned Models (up to 10 points).
Integrate your IK system with an existing animation system or game engine (such as Unity3D) in a way that allows the IK to fully control a skinned character. Note that you must use your own IK solver and not rely on any built-in solver, planning, or geometry code provided by the system.

IK + Character Animation (up to 10 points).
Integrate your IK system with an existing animation system or game engine (such as Unity3D) in a way that shares control with the built-in animation system. For example, the game engine may control the character's walking cycle, but your IK solver controls the arms to reach out and grab specific objects. Document carefully which part you animate and which part is built-in.

Part 2 - Challenge: Crowd Simulation [up to 60 points]
An additional challenge simulation is <u>required</u> if you are a graduate student, and optional for undergraduates. If you complete the challenge, you have a 72-hour extension on the project.

Flocking (up to 10 points (grad), 10 points (undergrad)).
Have multiple agents follow local interaction rules to generate group behaviors with emergent, animal-like motion. To receive full credit, the agents should be rendered in a way that provides a clear indication of their orientation. They should move and rotate smoothly and move through the environment in real-time without any unnatural hesitation or large changes in speed. The resulting motion should look natural and resemble animal-like motion.

Single Agent Navigation (up to 30 points (grad), 20 points (undergrad)).
Simulate a 2D agent moving through a cluttered 2D environment. The physical extent of the agent should be represented by a bounding geometry (e.g., bounding circle or bounding rectangle). As the agent moves through the environment, its bounding geometry should not overlap with any of the obstacles, and the agent should successfully navigate to its goal. To receive full credit, the agents should be rendered in a way that provides a clear indication of their orientation, should move and rotate smoothly, and should navigate through the environment in real-time without unnatural hesitation or large changes in speed.

Anticipatory Collision Avoidance (up to 20 points (grad), 10 points (undergrad)).
Simulate multiple agents in a shared space navigating among obstacles. Each agent should use an anticipatory crowd simulation technique to avoid obstacles, move towards their goals, and avoid collisions with each other in a smooth and natural

fashion. <u>To earn full points, the agents must demonstrate smooth, anticipatory, collision-free motion in *at least two interesting scenarios*.</u>

Full Crowd Simulation (up to 60 points (grad), 30 points (undergrad)).
Have multiple agents use a map, mesh, or tree to plan collision-free paths that move all agents to their goals in a shared environment, while also avoiding collisions with each other and the obstacles in the environment. You can use any method you prefer for collision avoidance (e.g., boids, social forces, TTC force) and path planning (e.g., PRMs, RRT, NavMeshes). <u>To earn full points, the agents must demonstrate smooth, anticipatory, collision-free motion in *at least two intriguing scenarios*.</u> Therefore, select a technique that can yield favorable outcomes.
[*Note: you can either get points for the full crowd simulation or the components of single-agent navigation and anticipatory collision avoidance, but not both]*

Art Contest
If you generate a pretty image (even by accident), save it to submit to the class art contest. A pool of honorable mentions will be given 2 points, and the grand winner gets 5 points. All winners will be chosen *completely subjectively*.

Project Report & Video* (10 points).
Your submission must be in the form of webpage with:

- Images of your physical simulations
- A brief description of the features of your implementation and timestamp of where they occur in your video(s).
- Code you wrote
- List of the tools/library you used
- Brief write-up explaining difficulties you encountered
- One or more videos showcasing features of your simulation
- Submission for the art contest (optional)

These 10 points for the submission itself will be based on the clarity of expression of the report, and to the degree which it quickly communicates what you tried, what worked well, and what didn't.

<u>Additionally, each feature you expect to get credit for must be documented in your submission videos in a way which clearly shows the resulting behavior</u>. If you do not show a feature in your submission video(s) you will not receive credit for it.

Grading Criteria
Simulations must animate well and look convincing to get full credit. Partially implemented features will receive partial credit. Points past those needed for full credit will count as extra credit, though at a discounted rate (see Scoring below). If you do other things that you think are cool and worth credit let me know beforehand and be sure to document them in the report.

Use of other code and tools
Anything you are getting credit for must be code you wrote for this course. You must write the code for the simulation yourself! External libraries may be used for aspects that are not related to simulation (e.g., rendering, camera motion, video capture) just be sure to document that you used these.

Partners & Groups
You are strongly encouraged to work in pairs for the project. Each pair should turn in only one assignment. Both people will be given the same grade. <u>You cannot repeat the same partner from a previous project, nor can you use the same partner in HW3.</u>

Project Scoring
Undergrads need 100 points for full credit, though you may choose to submit up to 120 points of work, subject to the following limits:
`110` for part 1 (IK)
`30` for the challenge (planning & crowds)
`10` for the report
… if you submit more than the limit, we will grade a random subset.

Graduate students need 120 points for full credit, though you may choose to submit up to 140 points of work, subject to the following limits:
`110` for part 1 (IK)
`60` for the challenge (planning & crowds)
`10` for the report
… if you submit more than the limit, we will grade a random subset.

Partial credit will be given. Scores computed as follows (points above 100 possible):
-*Undergraduate*: Grade is √(totalPoints * 100) [e.g., 100 points will be full credit]
-*Grad students*: Grade is √(totalPoints * 84) [e.g., 120 points will be full credit]

*Extra credit will only be given to assignments with at least an A- on the required features.