Final Report

ETHZ Game Programming Lab 2008

Bernhard Buss

Claudia Kuster

Jürgen Fornaro

1. Summary of Game Play

Momentum is inspired by the original board game where you have to balance a wooden board in order to move a metal ball from one end of a labyrinth to another, avoiding the holes along the path. The game provides two different game modes:

Classic game mode

Only single player mode. Simulates the original game play on the wooden board. The player controls a metal ball by balancing the board with the game pad. If the ball falls off the board, the game is lost.

Illustration 1: Classic mode, wooden board and metal ball. The board is rotated by the player.

Momentum game mode

Single player or multi-player modes (up to 4 players, single screen, see illustration 1).
The player directly controls his ball (a spherical cage containing a quantum particle) with the game pad, accelerating it in the desired direction (the board is fix). The particle

gains momentum if the player is able to navigate the ball smoothly through the labyrinth on the board, staying off the obstacles and the walls. The player can release the momentum of his quantum particle to perform special moves such as speed-ups, jumps or pushing an opponent off the board. The player that reaches the other end of the labyrinth, the so-called warp core matrix portal, wins.

An important feature of our game is universal (space-like) physics and rendering engines, enabling the construction of diverse game boards by simply importing a geometric model that was built using external software (e.g. Blender or Maya, in our case the software used for 3D content creation).

Illustration 2: Momentum Mode, each player controls a ball

2. Progress since Alpha Release

Menu Transitions

In order to bring the previously rather static looking menu alive, we added pushable buttons, designed transitions between the different menu screens and created a nice background particle effect for moving dust and stars. The buttons now give a feedback to the user that the button has been pressed and to other viewers which button has been pressed. In addition to that, when selecting a level, the board is zoomed in and the game starts with the same view (after a short loading time). Symmetrically, when leaving the game the board is zoomed out and the select level screen is faded in. This gives the players an impression as if all the levels and the menus are situated next to each other and located in one universe, as it should be. And for observers it is much more intuitive and pleasant to watch.

For the classic mode we planned and designed a transition which zooms into a cloudy planet, and then fades to the background of the classic mode (white clouds). But unfortunately, this transition was not fully ready for the final release.

Crystal Items

To give the players an additional source of fun, we added the gathering aspect to our game: Players can pick up crystals which fill up their momentum energy and also give 1 point to their total score. The ranking keeps track of how many crystals were collected by each player, to further lurk out the gathering behaviour of the players.

Furthermore we used the items as a balancing method for various levels where alternative routes can lead to the finish. Longer routes now get more attractive if there is a collectable item on the way to the finish. Also the additional momentum a crystal provides can lead players to different strategical thinking when playing a level and determining the route to the finish.

Difficulty levels

When connecting the controllers in the connect controllers screen, each player is allowed to individually select one of the three difficulties easy, medium and hard. These difficulty levels define the amount of momentum gathered when rolling around on the game boards, and how much a player looses when hitting a wall. In addition to that we introduced a helping force for the easy and medium difficulty, which breaks any velocity in a direction the player is not accelerating towards. This makes maneuvering much easier, and allows new players to have a chance against veterans. The selected difficulty is displayed in the

ranking, but has no further influence on the score of the players.

Threads

Multi-threading was introduced to improve performance on the Xbox. Since there are 3 CPUs, and the third one is unused by default, we decided to run two threads on the two hardware threads available on the third core of the Xbox CPU. One thread takes care of updating the particle system during game play, which is expensive because of the collision detection with the level, and the other thread is used for parallel loading of game resources, which also helps in eliminating/reducing the waiting time between menu transitions and when selecting a level.

Deployment on Xbox

As mentioned above we managed to successfully deploy our game on the Xbox. The performance problem of the particle system was solved by using the third core of the Xbox CPU. Also we had to reduce the render target size for the shadow mapping of the balls, and disable the soft shadowing for the balls. The global point light still casts nicely softened shadows, but for the points lights of each ball it got too expensive. Since we think this was one of our biggest successes, we feel very happy to have our game running on the Xbox finally!

3. Experience during the class

Overall, we are very happy with the final result of our project. Momentum is fun to play, it looks nice and it runs on the Xbox. Furthermore we were able to bring the game to a somewhat finished and polished state, filled with many little details that seamlessly add to the players game experience. We consider the project a success because we were mostly able to follow the project plan and reach our milestones. Significant changes to the initial concept were not necessary. The intermediate presentations and reports turned out to be very helpful for not losing track of the project plan, and also added to the overall experience on project management and presentation we acquired during the class.

4. Personal Impression of the Course

The course met our expectations because we knew in advance that it would be a lot of work and that the time would be short. Thanks to the help of the assistants and other course participants we never ran into serious problems that put the completion of the project at risk. But obviously one semester is a very short time for such a large scale

project.

In the lectures we would have liked to learn more about shader programming and the possibilities of GPU programming earlier in the course. We didn't know anything about it in the beginning and realized too late what great potential the GPU offers. The XNA framework proved to be easy to learn and fun to play with. Also we were able to develop our prototype very quickly mainly thanks to the XNA framework, thus making other prototype languages needless.

Looking back, we are proud of what we accomplished and appreciate everything we learned during the Game Programming Lab.

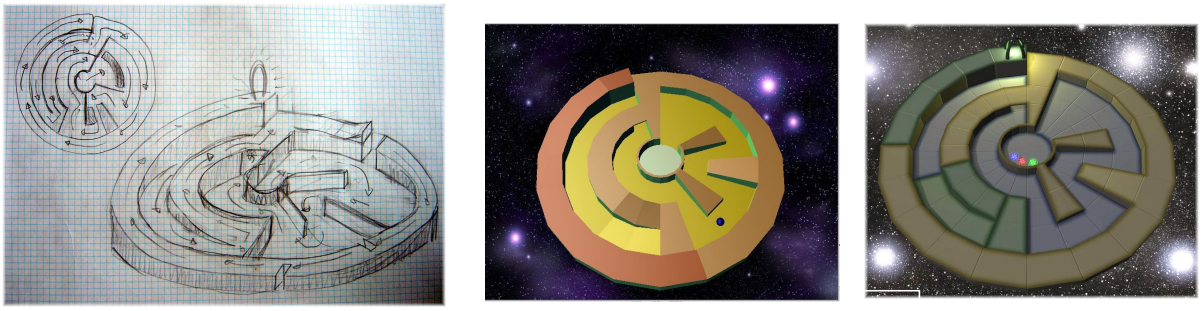

Illustration 3: Development process of our toughest level, and the lighting engine. For the latest screenshot, see front page.