# Milestone3: Stabilizing and Optimizing Physics Engine

In M3, I implemented Baumgarte Stabilization to prevent physics bodies from further penetrating one another. Additionally, two slope parameters are utilized to mitigate jittering when both speed and penetration depth are low. Currently, our system maintains only a single contact point for each manifold, which complicates the full stabilization of the boxes; ideally, four points should be maintained in total to maximize the area covered by these points. Consequently, during the demonstration, you may observe that the boxes exhibit some jittering—this behavior is expected.

文本

描述已自动生成

文本

描述已自动生成

Another optimization implemented is multithreading. In each stage of the physics update substep, I utilized a thread pool to efficiently calculate collisions and apply updates. This approach enables the physics system to iterate at a consistent speed while managing numerous objects simultaneously. Consequently, in the demo video, you can observe many boxes colliding and penetrating one another, all while maintaining stable FPS. This represents a significant performance enhancement.文本

描述已自动生成

文本

描述已自动生成 文本

描述已自动生成

文本

描述已自动生成

文本

描述已自动生成

The demonstration features a simulated coin pusher, utilizing cubes in place of actual coins, as the GJK and EPA algorithms are indifferent to the types of colliders used, provided that appropriate support functions are implemented. Players can manipulate the push bar to propel coins onto the stage while simultaneously generating an increasing number of coins within the scene. Observing numerous coins being pushed is quite satisfying; thus, a scoreboard has been incorporated to track the total number of coins successfully pushed off the platform.

工程绘图

中度可信度描述已自动生成  
  
Hope you enjoy the demo!