

컴퓨터애니메이션실습(CAL)

HW12_Mass-Spring Systems



Self-scoring table

	P1	P2	E1	Total
Score	1/1	1/1	1/1	3/3

2018707068 김경환

KwangWoon University

Code Analysis:

먼저 이번에는 Particle System Dynamics에 Spring force를 추가함으로써 Mass-Spring System을 구현한다.

해당 과제에서는 external force fields로 Gravity과 Spring force를 부여하였고, 4개의 벽을 구성하여 이에 대한 Collision Handling을 수행하였다.

그리고 이러한 Elastic Spring을 기술하기 위해 Stiffness constant: k 와 Initial spring length: l , Current spring length: L 을 사용하여 Hooke's law에 따라 Spring force의 크기를 계산하였다.

이때 Spring force는 edge(=spring)에 대하여 양 끝점에서 서로 반대방향의 힘이 작용하므로 한 점에 대해 힘의 방향을 계산한 후 다른 한 점은 부호를 바꿔 적용해주면 된다.

또한, spring constant(=Stiffness constant)는 방향키로 조절이 가능한 Global spring constant를 주고 길이에 반비례하게 각 edge에 대해 계산하여 부여하였다.

총 4개의 point와 3개의 edge(=spring)에 대해 실험이 진행되며 이는 직접 초기에 particle의 위치와 속도, rest length를 부여해준다.

이러한 rendering을 update하기 위해서는 먼저 Total force(internal force, external force)를 계산하고 시간 t 에 대한 position과 velocity를 계산해야하는데, 이는 상미분방정식을 사용하여 적분을 수행해 구해낸다.

해당 system의 force는 2차로 표현되기 때문에 두 개가 결합된 1차 상미분방정식으로 표현하여 계산한다.

External force로 작용하는 gravity는 total force에 더해주는 방식으로 적용하고, internal force인 spring force는 edge의 current length와 edge의 end point에서의 힘의 방향을 계산하여 위에서 설명한대로 Hooke's law에 따라 값을 계산하여 두 end point에서 서로 방향이 반대가 되도록 한 뒤 더해줘서 부여한다. 여기서 spring damping도 이와 같은 방법으로 하나의 end point에서 상대적인 힘의 작용 방향과 current length를 계산하고 damping constant를 곱해서 빼주는 방법으로 부여한다.

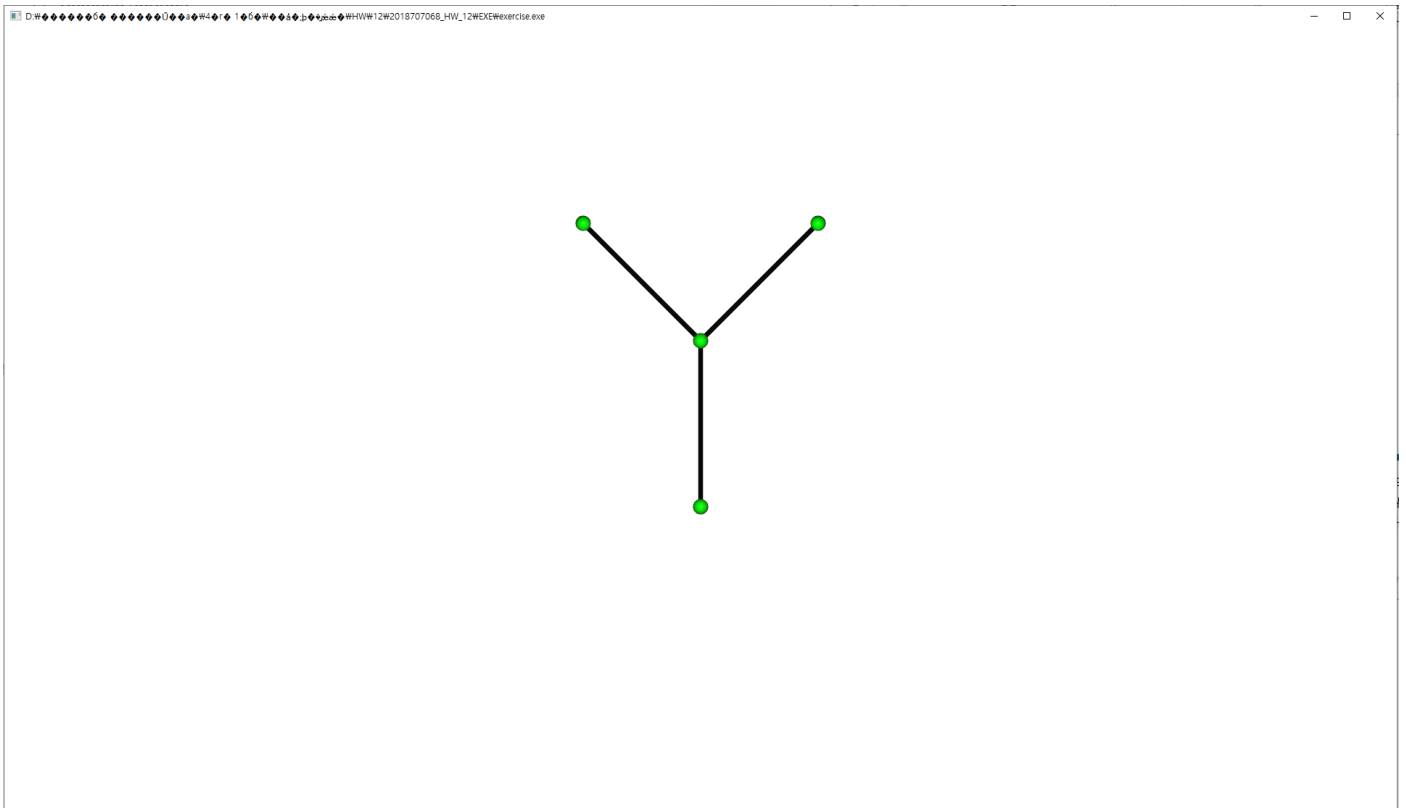
여기서 적분을 수행하는 방법으로는 두 가지를 사용하였는데, 먼저 가장 간단한 Euler's method는 현재의 position과 velocity를 사용하여 time step 다음의 position, velocity를 계산해낸다.

그리고 Modified Euler's method는 먼저 현재의 velocity를 사용하여 다음 time step의 velocity를 구하고, 이를 반영하여 다음 time step의 position을 계산한다.

Euler's method는 Sub steps의 개수에 의존적이므로 Sub steps의 개수가 적을 때는 만족스럽지 않은 모습을 보인다.

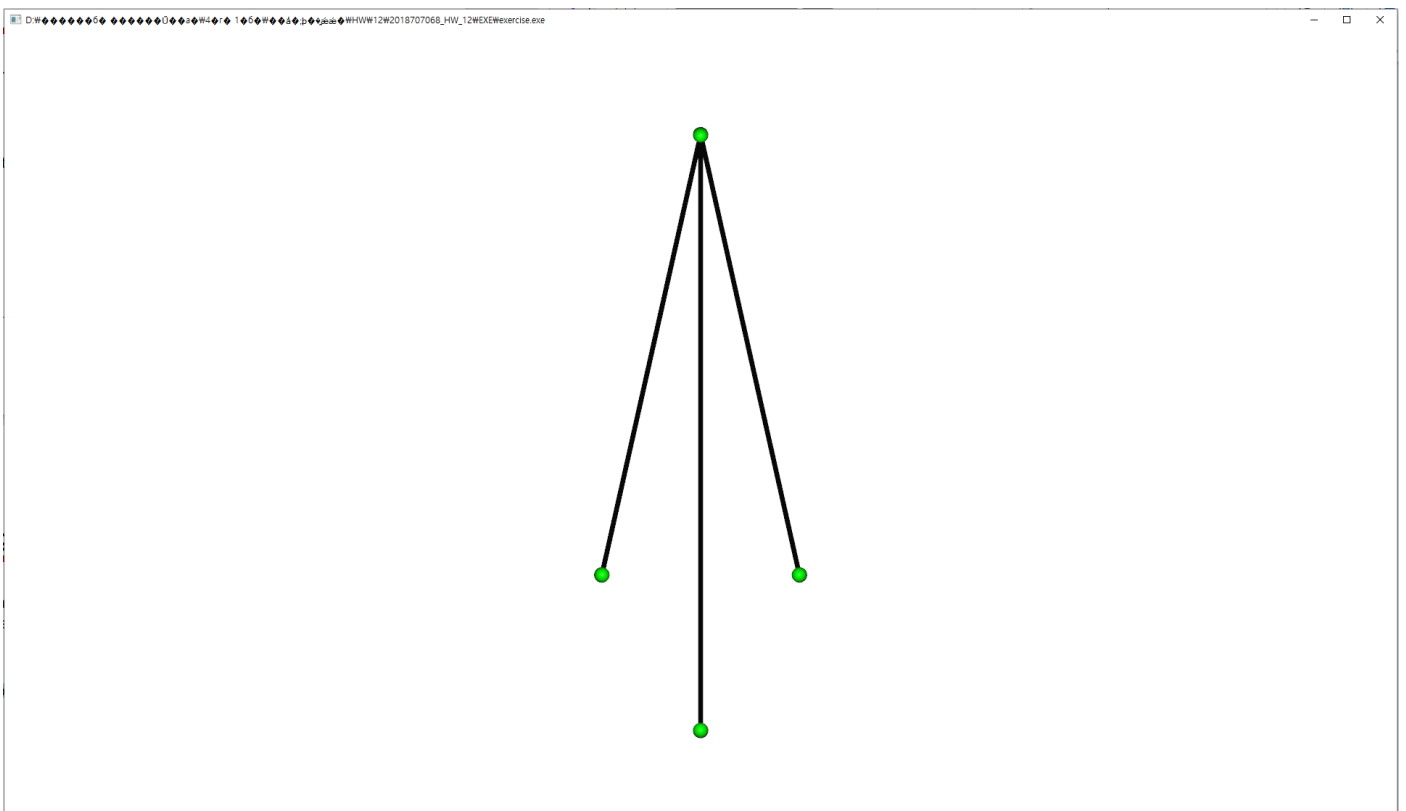
하지만 Modified Euler's method는 그에 비해 Sub steps의 개수에 덜 영향을 받아 Sub steps의 개수가 적어도 상대적으로 만족스러운 모습을 보인다.

벽에 대한 collisionHandling은 직전 과제의 보고서에서 상세하게 설명했으므로 여기서는 넘어가겠다.



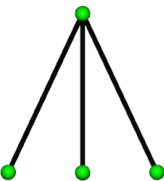
앞으로 나올 Practice 01, 02, Exercise 01의 Particle 초기 위치이다.

Practice 01. Time integration with Euler's method:



Sub step의 개수로 1, GobaI Spring constant로 1, Damping coefficient로 0.01을 주고 Point Damping을 수행하며 Euler's method로 상미분방정식을 풀었을 때의 모습이다.

Point damping을 수행함에도 시간이 지남에 따라 Spring이 rest length를 유지하지 않고 이상해지는 것을 볼 수 있다.



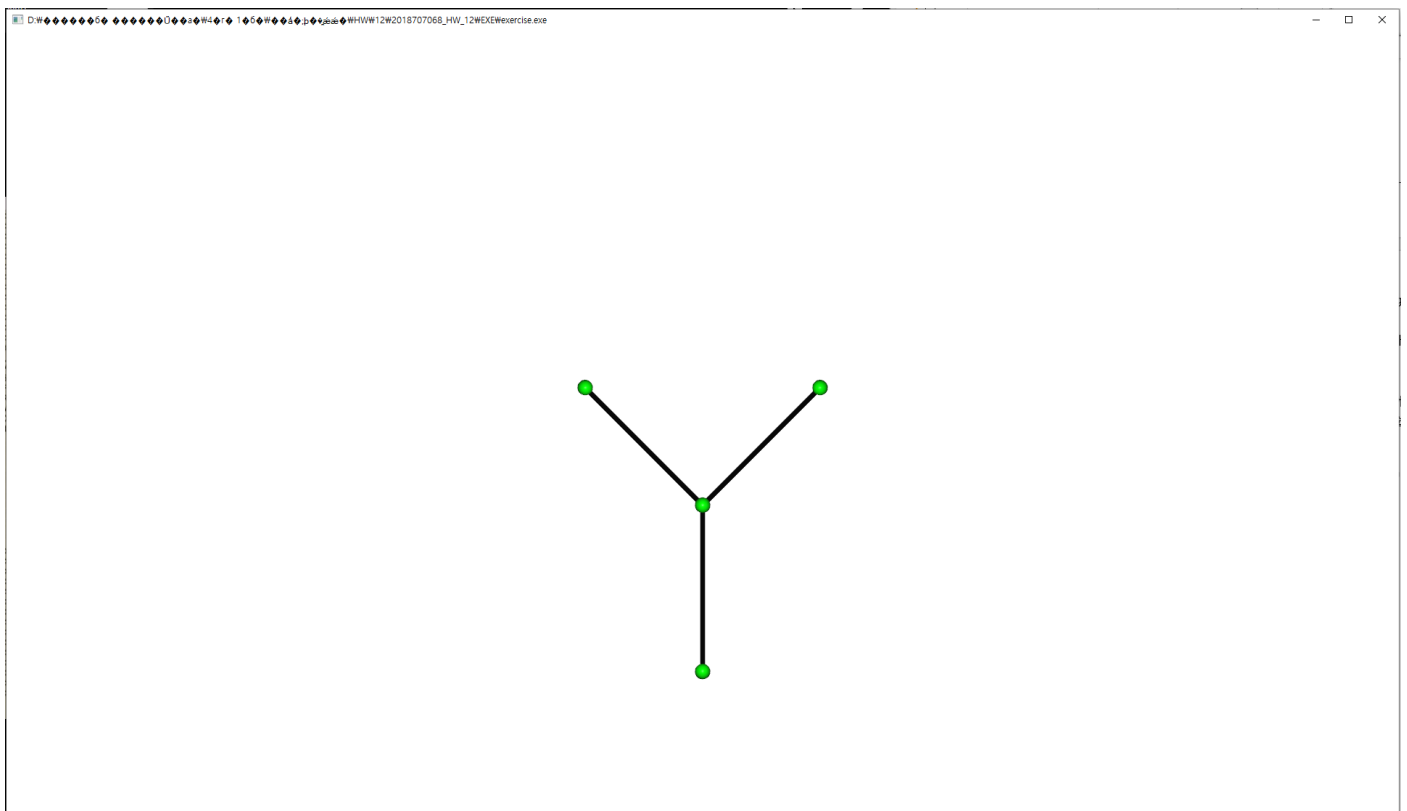
Sub step의 개수가 1일 때에 비해 더욱 안정적인 만족스러운 결과를 보이는 것을 볼 수 있다. 이는 1 frame에 소요된 시간을 Sub step의 개수인 9로 나눴기 때문에 오차가 크게 줄어들었다.

Sub step의 개수로 1, Goba Spring constant로 1, Damping coefficient로 0.01을 주고 Point Damping을 수행하며 Modified Euler's method을 수행했을 때의 모습이다.

Modified Euler's method는 Euler's method와 달리 먼저 현재의 velocity를 사용하여 다음 time step의 velocity를 구하고, 이를 반영하여 다음 time step의 position을 계산하기 때문에 Euler's method보다 만족스러운 결과를 보이는 것을 확인할 수 있다.

그리고 Modified Euler's method는 Sub step의 개수가 1일 때에도 상당히 만족스러운 결과를 보여주기 때문에 Sub step의 개수를 키워도 큰 차이를 육안으로 확인하긴 쉽지 않다.

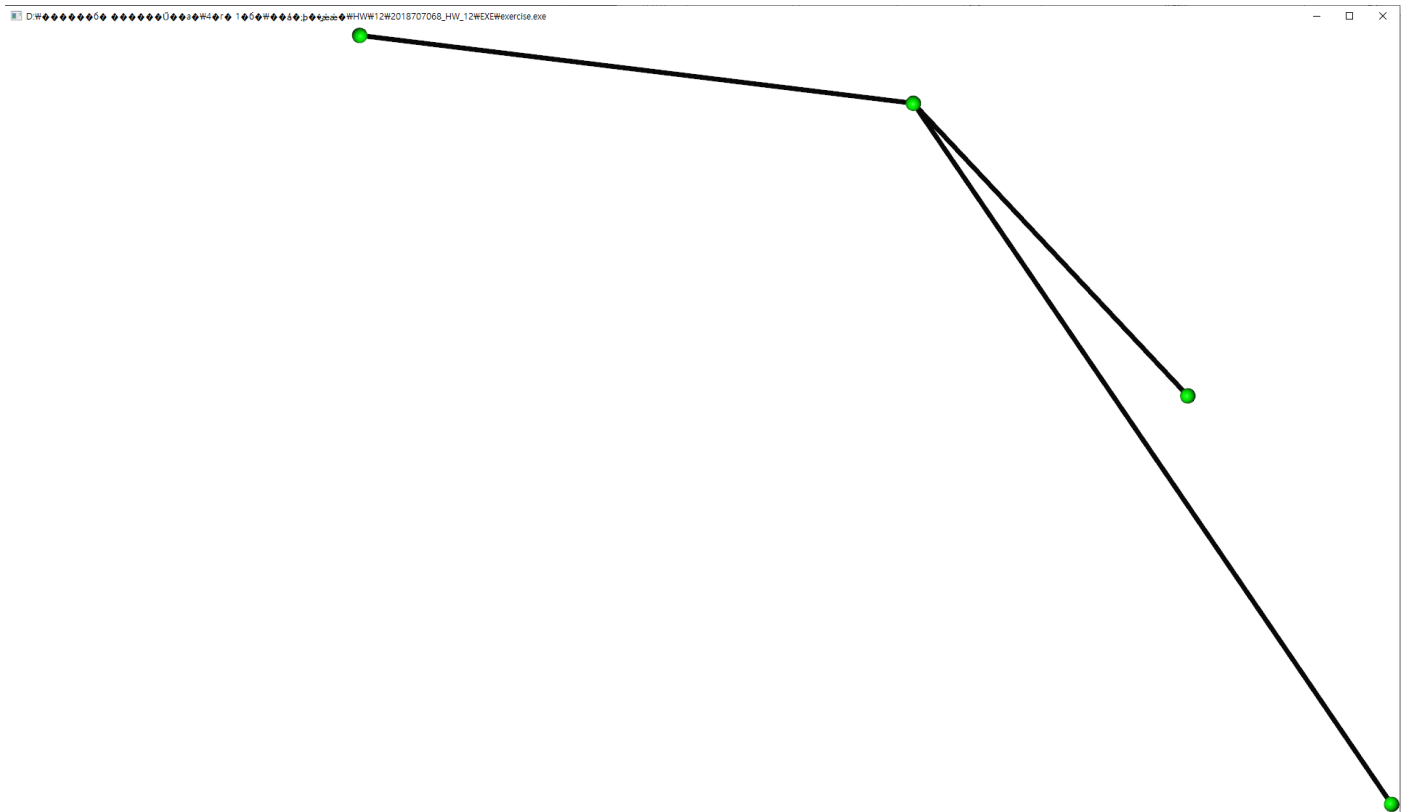
Exercise 01. Point damping vs. damped springs:



Sub step의 개수로 1, Goba Spring constant로 1, Damping coefficient로 0.1을 주고 Point Damping을 수행하며 Euler's method로 상미분방정식을 풀었을 때의 모습이다.

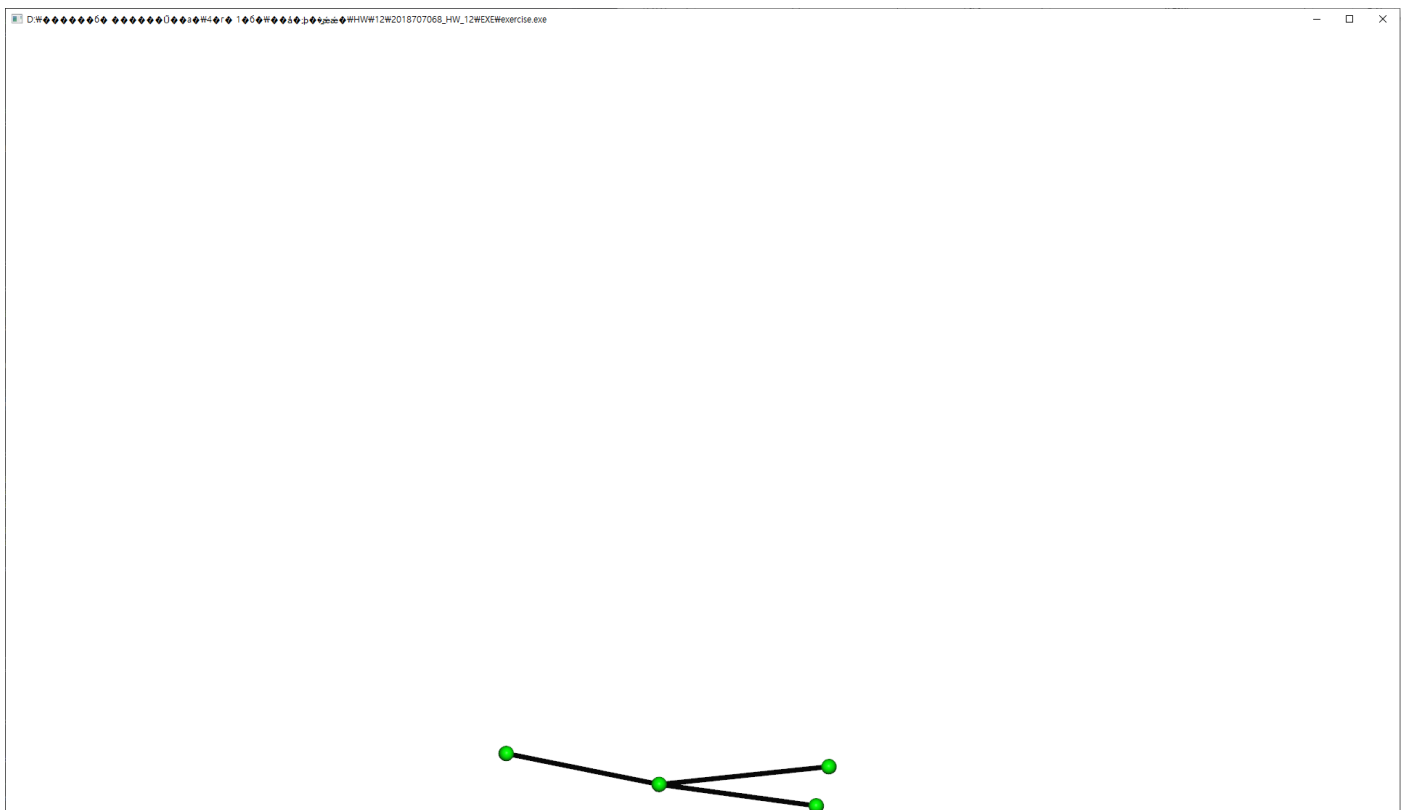
위 practice1에서처럼 Damping coefficient를 0.01로 낮게 줬을 때는 Point damping을 수행함에도 시간이 지남에 따라 Spring이 rest length를 유지하지 않고 이상해지는 모습을 보였으나 Damping coefficient를 0.1로 높이니 external force인 gravity에 damping이 걸려 천천히 떨어지는 모습을 볼 수 있다.

Point Damping은 Mass Spring System에서 어떠한 Damping coefficient를 부여해도 만족스럽지 않다.



Sub step의 개수로 1, Gobal Spring constant로 1, Damping coefficient로 0.01을 주고 Spring Damping을 수행하며 Euler's method로 상미분방정식을 풀었을 때의 모습이다.

Spring Damping을 수행하지만 Damping coefficient를 0.01로 낮게 부여하여 시간이 지남에 따라 Point Damping처럼 Spring이 rest length를 유지하지 않고 이상해지는 모습을 보인다.



Sub step의 개수로 1, Gobal Spring constant로 1, Damping coefficient로 0.1을 주고 Spring Damping을 수행하며 Euler's method로 상미분방정식을 풀었을 때의 모습이다.

Damping coefficient를 0.1로 높이니 internal force인 spring force에 damping이 걸려 Spring이 rest length

를 유지하지 하며 만족스러운 모습을 보이는 것을 확인할 수 있다.

여기서 spring을 형성하는 point particle이 바닥에 붙지 않고 통통 튀는 것을 볼 수 있는데 이는 Sub step의 개수로 1주고 Euler's method로 상미분방정식을 풀었기 때문이다.

이로 인해 벽을 collisionHandling 하는 과정에서 세밀한 계산이 안돼 튀어오른 것을 볼 수 있다.

이처럼 Spring Damping은 Mass Spring System에서 적절한 Damping coefficient를 부여하면 만족스럽다.