



## SW프로젝트 제안서

프로젝트명	3D Path Finder where gravity works	
프로젝트 유형	지정 주제 (    ) 자유 주제 (    )	졸업 작품 (    ) 졸업 논문 (    )
프로젝트 요약	현재 상용화되고 있는 대부분의 path finder는 2차원 평면에서의 길 찾기 시스템, 3차원 공간에서의 최단경로 등이다. 이에 더해 중력이 작용하고 지형지물을 이용할 수 있을 때의 path finder를 Reinforcement Learning으로 구현해보고자 한다.	
R&D 산출물	SW (    ), HW (    ), 특허 (    ), 논문 (    ), 프로그램등록 (    )	
지도교수	권태수	
예상기간	2021.03.13. – 2021.08.31.	

전공	학번	학년	이름	연락처
컴퓨터소프트웨어	2014004211	4	표영권	xinkis96@gmail.com 010-6282-9610
컴퓨터소프트웨어	2015004984	4	정동훈	dhjeong081@gmail.com 010-6655-6201

- 연락처는 이메일과 전화번호를 모두 쓰되 반드시 수신 가능한 것으로 기입  
(연락을 받지 못해 불이익을 당할 수 있음)



## 목 차

1. 프로젝트 배경 및 목표
2. 프로젝트 주요 내용
3. 추진 계획
4. 결론
5. 참고 문헌

## 1. 프로젝트 배경 및 목표

### 가. 프로젝트의 정의

본 프로젝트는 중력이 작용하여 위, 아래 방향으로 path를 결정할 때에 더 cost가 부가되는 보다 현실적인 system에서의 path finder를 의미한다. 만약 지형지물이 어떤 특정 각도 이상의 기울기를 가졌다면 그것을 오르기 위해서는 stamina가 소요된다. 또한 높은 곳에서 활강을 할 때에도 stamina가 소요된다. stamina는 시간이 지나면 회복되는 시스템이며, 회복하기 위해서는 제자리에서 일정 시간 가만히 있어야 한다. 이때 자유낙하도 포함된다. 하지만 일정 높이 이상에서 떨어져서 지면에 착지했을 때에는 피해를 입게 되므로, 피해를 입지 않으면서 최단 시간에 목적지까지 도달하는 path finder system이다.

### 나. 프로젝트의 기술적 배경

기본적으로 Reinforcement Learning은 agent가 environment에서, 어떤 state에서 action을 취함에 따라 reward가 얻어지고 다음 state로 진행되는 학습 방식을 말한다. Reinforcement Learning의 한 예로 그래프 탐색 알고리즘 중 하나인 Dijkstra 알고리즘이 있다. 출발 state에서 목표 state까지 가장 적은 cost를 사용하여 도달하는 탐색 알고리즘이다. 이를 응용한다면 설정된 environment에 대하여 경로(action)에 따라 cost를 부여함으로써 이 값들의 합이 최솟값이 reward가 될 수 있도록 system을 설계하고 학습시키고자 한다.

### Typical RL scenario

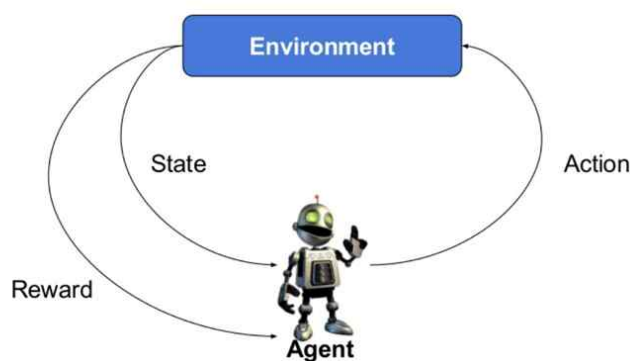


그림 1 Reinforcement Learning의 기본적인 시나리오

상용화되어 있는 3D path finder의 경우, agent에 외력으로 작용하는 것이 없는 조건에서 3D 공간에서의 최단 경로를 찾는 기술이다. 본 프로젝트에서는 여기에 중력이라는 외력과 강한 충돌 가능성을 염두를 해야 하는 environment를 설계하고자 한다.



#### 다. 프로젝트의 필요성

우주의 외계 행성을 탐사하는 로버, 남극이나 심해 등에서 탐사를 하는 로봇 등이 탐사를 하는 과정에서 장애물을 마주치거나 예상되는 정보가 있을 때 단순히 장애물을 우회하는 것이 아니라 이를 극복하고 넘어가는 것이 더 cost가 적게 소요될 가능성이 있다. 이 경우 소체의 크기나 소체가 버틸 수 있는 내구도 등의 정도에 따라 달라질 수 있으므로 본 프로젝트에서는 사이즈와 각 소체별 stamina 보유량을 다르게 적용시켜 이를 시뮬레이션해보고자 한다.

#### 라. 최종 목표

agent 소체가 언덕, 절벽, stamina를 사용하지 않는 경사진 길, 평지 등 다양한 3차원 환경 속에서 주어진 출발점과 목적지 간에 cost를 최소화하는 경로를 찾는다.

#### 마. 세부 목표

- 1) agent가 피해를 입어 행동불능상태가 되지 않도록 한다.
- 2) stamina는 회복될 수 있으므로 시간 cost를 최소화하는 것을 목표로 한다.
- 3) 평지만을 골라서 가는 길이 아니라 어느 정도 절벽을 기어오르거나 활강하여 도달할 수 있는 길이 있음을 보인다.
- 4) 절벽 등을 기어오르는 도중에 도약하여 활강을 할 수 있다.
- 5) 필요할 경우 떨어지면 행동불능이 되는 높이, 지면의 조건 등을 학습한다.



## 2. 프로젝트 내용

### 가. 제반 사항

- 1) 평지, 절벽 등 어디서든 제자리에 가만히 있으면 stamina는 회복되지만 시간은 흐른다. 평지 및 낮은 각도( $\theta$ )의 경사로를 이동할 때의 표준 속력(Standard Velocity)을 가지는 동안에는 stamina를 소모하지 않으며 이 때에도 stamina는 회복된다.
- 2)  $\theta < \gamma \leq \frac{\pi}{2}$ 의 기울기를 가진 벽을 기어오를 때에는 stamina를 소모한다. 아직 벽에 있을 때 stamina가 0이 되면 그대로 추락한다.
- 3) 추락했을 때 소체가 버틸 수 있는 높이라면 추락을 시도할 수 있다. 이를 바탕으로 소체의 높이와 소체 바닥에 착지가 예상되는 object 사이의 거리를 계산할 수 있어야 한다. object에 안전하게 착지해야 하므로 항상 지면으로부터  $\theta$  각도 이내의 평면에만 착지하도록 한다.
- 4) 평상시 평지와 낮은 각도의 경사로를 이동할 때에는 표준 속력을 갖고, stamina를 소모하여 질주 속력(Dash Velocity;  $dv = a \cdot sv, a \in R$ )으로 가속할 수 있다. 첫 가속 시에는 설정된 IA(Initial Accel)값만큼 stamina를 소모하여 질주 속력까지 도달한 후(이 때까지 걸리는 시간을  $t_a$ , 그 이후에는 초당 일정한 값만큼 stamina를 소모하며 질주 속력을 유지할 수 있어야 한다. stamina를 모두 소모한 후에는 속도가 점점 감소하여 표준 속력으로 유지된다. 이 또한 감속하는데 걸리는 시간은  $t_a$ 로 동일하다.
- 5) 활강을 할 경우 질주 속력보다는 느리지만 표준 속력보다는 빠른 속력으로 천천히 전진할 수 있다. 이 경우에도 stamina는 소모되며, 제자리에서 천천히 활강도 가능하다.

### 나. 사용 기술

Unity에서 제공하는 Nvidia 사의 PhysX engine을 이용하여 물리 엔진 시뮬레이션 환경을 조성한다. 또한 agent로 사용될 component는 Rigidbody를 이용하거나 시중에 프리소스로 사용 가능한 모델(사람 형태 등)을 사용해보고자 한다.

Rigidbody component의 경우 mass, using gravity option, collision detection 등의 기능을 제공한다. 자유낙하의 경우 갈릴레오의 자유 낙하 법칙이 적용되어, mass 값에 상관없이 항상 같은 가속도로 떨어지게끔 설계되어 있다.

agent object의 충돌 컴포넌트(Collider)를 Capsule Collider로 지정하면 사람 형태의 agent 소체에 적용 가능하고, Box Collider로 지정하면 로버와 같은 로봇 형태의 소체에 적용 가능하다. 이를 통해 지형과 충돌 판정을 판별할 수 있다.

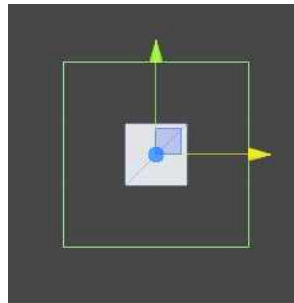


그림 2 Box Collider

[그림 2]의 노란색 네모 박스가 Box Collider의 형태와 크기이다. 해당 영역에서 다른 Collider와의 충돌을 판별한다. 지형지물의 경우 고정된 Collider이므로 Rigidbody를 추가하지 않고 static Collider component만 추가하여 리소스를 절약할 수 있다.

#### 다. 현재 기술의 이용 및 연구개발 추세와 계획

##### 1) Mars Pathfinder project

실제로 최근 화성 착륙에 성공한 Perseverance호의 경우, 총 개발 기간 3년 이내, 개발 비용 1억 5천만 달러 이내, 다른 행성에서 간단한 통신체계로도 정보 송수신이 가능한지를 확인, 발사 비용과 명령 조달 비용을 포함하여 프로그램에 투자되는 총 비용을 2억 8천만 달러 이내로 제한한다는 NASA의 공약을 실현하며 화성 표면에서 로버의 기동성을 체크, 실용성이 있는지를 증명한다는 목표 하에 진행중인 프로젝트이다.

Mars Pathfinder의 경우 값싼 비용으로 화성 탐사의 임무를 달성하고 정보의 용이한 송수신이라는 목표 하에 화성을 탐사하기 위한 프로젝트이고, “더 빠르고 더 뛰어나며 더 값싼”이라는 구호에서 드러나는 optimal goal은 본 프로젝트와 같다고 할 수 있다. Mars Pathfinder의 경우 화성의 불안정한 대기권에서 안전한 착륙을 위해 설정한 목표와 본 프로젝트에서 agent 소체가 파손되지 않고 stamina 즉 외력에 의해 소모되는 resource를 줄이고자 하는 목표가 같다. Mars Pathfinder의 경우 낙하라는 동일한 조건을 두었을 때 어느 시점에 어떤 대기의 조건에서 낙하산을 펴고 착륙을 위해 어떻게 가속 및 감속할 것인가를 정하는 것이 주요 연구 과제 중 하나였다.

기존의 성공 모델인 Curiosity 탐사 로버는 9년째 화성에서 활동 중인 모델이므로, 이 설계를 차용하면 비용과 시간을 크게 절감할 수 있다. Curiosity의 모델을 차용하고 여러 가지 기술이 발전된 탐사 기구와 새로운 목표를 가지고 Perseverance호는 발사되었다. 화성에 착륙하기 위한 가장 중요한 단계는 대기 진입 후 터치다운까지의 6분 50초에 있다. 지면으로부터 125km 떨어진 상공에서 초속 5.9km의 빠른 속도로 낙하하는 로버를 실은 우주선을 보호해주기 위해 최대 1300°C를 버티는 열차폐막을 설계하였으며 이는 낙하 진입시의 열을 막아주고 우주선의 속도를 줄여준다. 이후 4분 동안 자유 낙하를 한 끝에 낙하산을 펼치게 되는데, 기존의 우주선은 특정 속도에 도달하면 낙하산을 펴도록 세팅이 되어있지만 Perseverance호는 목표 지점과의 거리를 고려해 최적의 시기에 낙하산을 펼치도록 설계한다(Range-Trigger). 현재 속도와 목표 지점을 고려해 더 멀리 날아갈 것으로 예상되면 낙하산을 좀 더 일찍 펼치고, 반대로 덜 날아갈 것으로 예상되면 낙

하산을 좀 더 늦게 펼치도록 한다. 이후 열차폐막을 분리하고 착륙예상지점의 지형을 스캔하여 위험도를 측정하고, 필요할 경우 새로운 경로를 재지정해주는 시스템 (Terrain-Relative Navigation)을 도입한다. 이후 고도 약 1.6km 지점에서 낙하산과 백셀을 분리하여 제트팩의 도움으로 감속하여 Skycrane을 이용해 터치다운하도록 한다. 과거의 경우 에어백을 사용하여 터치다운을 하였으나, 무게가 약 1t에 달하는 Perseverance 호를 감당하기 위한 에어백을 싣는 것만으로도 비경제적이기 때문이다. 이때 아주 조심스럽게 착륙하기 위해 20m 지점에서 초속 0.75m의 속력으로 천천히 착륙하도록 한다.



그림 3 Skycrane, 자료 출처 Youtube ‘복툰’

## 2) 3D 게임에서의 path finder

현재 상용화되고 있는 3D 모델링 기반 게임에서는 아직까지 본 프로젝트와 유사한 path finder가 존재하지 않는다. 그래서 일일이 유저로 하여금 가능한 경로를 sequential 하게 지정하도록 한다. 이러한 예시로 Nexon 사의 ‘마비노기’라는 게임이 있다. 아래 [그림 4]를 보면 우하단에서 출발해 좌상단으로 가는 경로인데, 깃발이 새겨진 위치를 next goal position으로 보고 총 11개의 path를 거치는 것을 알 수 있다.

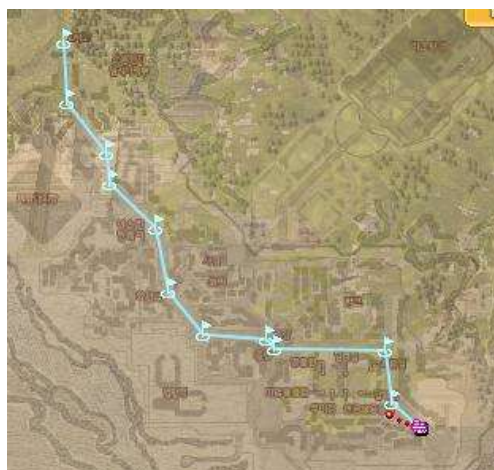


그림 4 Nexon 사의 게임 ‘마비노기’ 경로 지정

실제로 위의 경로는 육로를 향해 캐릭터가 걸어가는 경로이고, 해당 게임에서는 지형 지물을 이용해 벽을 타거나 활강을 할 수 없어 육로를 이용할 수밖에 없다. 이를 개선하

기 위해 해당 게임에서는 비행 펫을 이용해 하늘을 날아 쉽게 직선 경로로 활주할 수 있으나, 더 높은 산이 있으면 돌아가야 한다든지 시스템 상 제약이 다소 있다.

또 다른 게임 '원신'에서는 마비노기와는 달리 지형지물을 이용하여 등반할 수 있으며, 등반하거나 활강하는 동안 소모되고 있는 stamina를 호형 그래프로 보여준다[그림 5].



그림 5 캐릭터가 지형을 등반하거나 활강할 때 우측에 stamina gauge가 표시되며 action을 수행함에 따라 점점 줄어든다.

해당 게임에서는 아직 path finder나 경로 지정 기능이 구현되어 있지 않으며, 유저들이 직접 탐사하면서 최적의 경로를 찾아 기억하곤 한다.

게임 원신에서는 자유낙하를 하다가 도중에 다시 활강 모드로 변경함으로써 높은 상공에서 안전한 착지까지의 시간을 줄이고 동시에 자유낙하 시간 동안 stamina를 회복하기도 하는데, 이는 실제로 명왕성 탐사선에서도 적용되었던 방법이다. 우주공간을 비행하는 동안에는 관성으로 누적된 속도로 비행하면서 시스템을 계속 가동시켜두면 불필요한 전력소모가 많아지므로 이와 같이 특정 term을 두어 수면(게임에서는 자유낙하), 가동 및 점검(게임에서는 다시 활강 모드), 다시 수면 모드로 전환의 과정을 거쳐 전력 소모를 획기적으로 줄이는데 성공했다[그림 6].

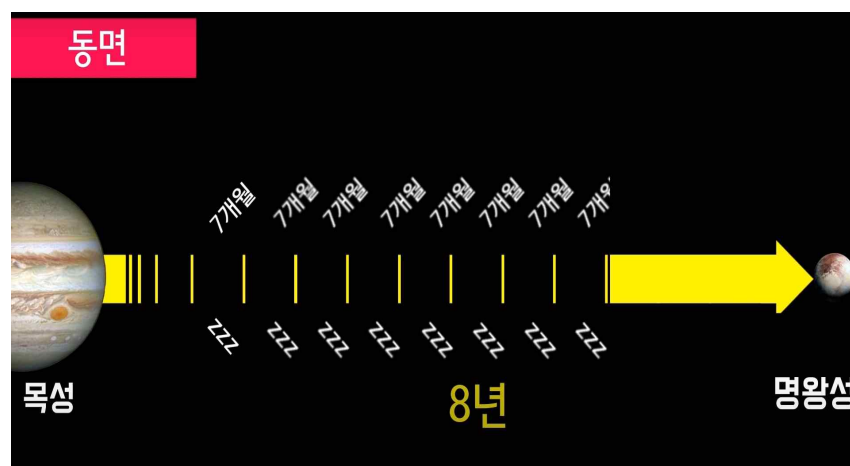


그림 6 뉴호라이즌스 호가 목성에서의 flyby를 거쳐 명왕성을 가는 동안 진행된 동면 시스템. 연료 사용을 획기적으로 줄였다. 자료 출처 Youtube, '북툰'





### 3. 프로젝트 추진 계획

팀 구성	역할	공통 임무
표영권	RL model 설계	설계한 RL model을 Unity 환경에서 실험하고 시뮬레이션 제작
정동훈	Unity 엔진 활용	
프로젝트 계획		
3월	Reinforcement Learning 설계 및 그래픽 지식, 기술 습득(프로젝트 끝까지 지속)	
4월	Unity engine 활용 스킬 익히고 시뮬레이션 환경 조성. 맵을 제작하고 stamina system 도입, 시간 측정 도구 탐색(시험 기간 등 고려)	
5월		
6월	Unity engine에서 agent 소체의 movement를 코딩한 대로 움직일 수 있도록 하는 기능 학습	
7월	RL modeling에 기반한 movement 기능 구현	
8월	실제로 실험해보면서 피드백 진행, 수정사항 반영하여 지속적으로 프로젝트를 수행할 수 있도록 함(세부목표 달성 및 최종 목표 달성)	
9월	결과를 문서화하여 마무리	

## 4. 결론

가. 최종 결과물에 대한 예상

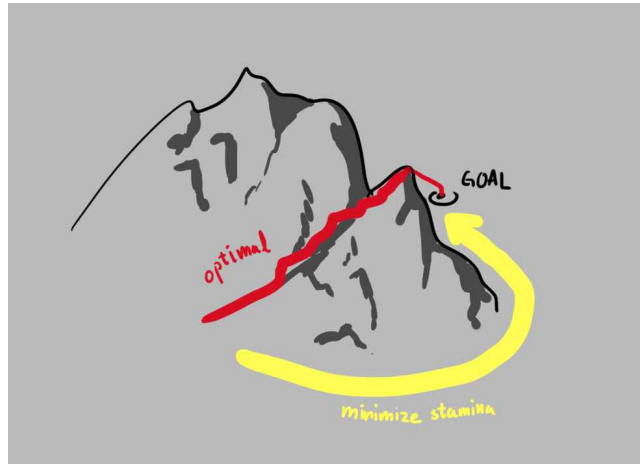


그림 7 optimal path(red line) and minimize stamina path(yellow line)

[그림 7]과 같이 단순히 지형지물을 우회하여 가는 경로가 아니라 그것을 활용하여 가는 optimal path가 있음을 보인다.

나. 향상될 것으로 기대되는 개인의 역량

- 실무 영역에서 자주 활용되는 Unity engine에 알고리즘을 설계한 코드를 탑재할 수 있는 스킬의 습득
- Reinforcement Learning에 있어서 중요한 FSA(Finite State Automata) 설계 능력
- 현실에서는 더욱 복잡한 문제를 좀 더 단순화한 게임 모델을 응용하여 문제를 단순화하고 해결하기 위해 탐구하는 능력

다. 응용분야와 학술적, 산업적 기대 효과

로보틱스 공학에 있어서 좀 더 지형지물을 이용하여 목적지에 도달하거나, 시각적으로 관측되는 object에 있어 사람과 같이 이를 이용할 수 있도록 설계하는 분야에 작게나마 보탬이 될 것으로 보인다. 자연의 지형의 경우에는 풍화와 퇴적과 같은 일련의 과정 끝에 생긴 결과물이므로, 로버 등의 탐사 로봇이 관측 불가능한 뒤편에 있어서 이미 학습한 map이 존재한다면 뒤편으로 가기 위한 최적의 경로를 알고 탐사를 진행할 것이다. 이로 인해 중간 과정에서의 연료나 자원 사용을 줄일 수 있어 보다 경제성을 기대할 수 있을 것이다.



## 5. 참고 문헌

- Conceptual introduction – <https://www.guru99.com/reinforcement-learning-tutorial.html>
- Real-Life Applications of Reinforcement Learning – <https://neptune.ai/blog/reinforcement-learning-applications>
- Unity 3D Pass Finder Asset – <https://assetstore.unity.com/packages/tools/ai/pathfinder-3d-100285?locale=ko-KR>
- Unity 3D basics – <http://blog.naver.com/PostView.nhn?blogId=nuberus&logNo=50183659187>
- NASA의 무인 탐사선 발전 동향 – <https://scienceon.kisti.re.kr/srch/selectPORSrchArticle.do?cn=NART69885036&dbt=NART, 2012>
- Reinforcement Learning – [https://en.wikipedia.org/wiki/Reinforcement\\_learning](https://en.wikipedia.org/wiki/Reinforcement_learning)
- Perseverance호에 대하여 – <https://www.youtube.com/watch?v=5k1gR5HtGpU>, 17.02.2021
- 뉴호라이즌스 호에 대하여 – <https://www.youtube.com/watch?v=3v9X3ufzBnQ>, 25.11.2020
- flyby - [https://en.wikipedia.org/wiki/Flyby\\_\(spaceflight\)](https://en.wikipedia.org/wiki/Flyby_(spaceflight))