고급 소프트웨어 실습

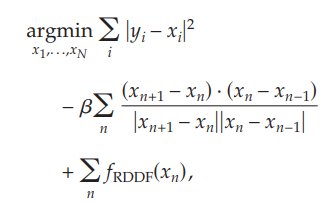
분반: 2

학번: 20181650

이름: 안도현

-path smoothing

RDDF는 경로에 대한 대략적인 묘사로서, 급커브가 포함되어 있기 때문에 RDDF waypoints를 기반으로 주행할 시 가속도를 주체하기 힘들어 위험할 수 있다. 따라서 그보다 부드러운 경로를 기본 궤도로서 설정해야 빠른 속도와 높은 정확도의 주행을 완성할 수 있다. 곡률 또한 실제 경로와 비슷하게 예측해야 측면 방향의 변화를 줄여 부드러운 주행이 가능하다. 따라서 이런 기본 궤도는 아래와 같은 과정을 거쳐 부드러워질 수 있다. 먼저 RDDF에 현재 곡률에 비례하여 점들을 추가한다. 그리고 이 점들은 least-squares optimization을 거쳐 더 부드러워진다. least-squares optimization이란 아래와 같은 수식을 통해 이루어진다.



x는 최적화될 점들, y는 RDDF의 anchor points이다. 우선 그들의 거리를 minimize하고, 두번째 항을 통해 점들 간 dot product를 줄여 이루는 각을 줄인다. 그를 통해 부드러운 곡선이 되는 것이다. 마지막 항의 f는 RDDF 경계로 가까워질수록 빠르게 커지는 함수로서, 부드러워진 곡선이 올바른 RDDF corridor 내에 존재하도록 만든다. 이 최적화 이후 단계로 경로의 미분 가능성을 위해 cubic spline interpolation을 거친다. 마지막으로 부드러운 궤도에서 각 waypoint에 대응되는 속도 제한을 계산한다. 이는 세 개의 값 중 최솟값으로 결정된다. 원래 RDDF에서 해당 영역에 적용되던 속도 제한, 횡방향 가속도 한계로 인한 속도 제한, 감속 한계 제약으로 인한 속도 제한이 그것이다. 두번째 값은 코너를 돌 때 올바르게 감속하도록 해주고, 세번째 값은 코너를 예상하여 미리 감속하도록 강제한다.

결론적으로, 주어진 figure을 보면 lease-squares optimization에서 베타 값을 어떻게 주느냐에 따라 부드러운 정도가 달라짐을 확인할 수 있다. 중요한 점은, 위에서 설명한 부분들이 원래의 RDDF 파일을 직접 수정하진 않는다는 점이다. 이는 RDDF를 활용해 단순히 좌표 시스템으로서 사용되어 올바른 경로로 나아가게 할 뿐이지, 레이스의 룰에 따른 특정 제약 조건을 만족시키는지를 별도로 파악하거나 하는 것과는 관련이 없다.