L4G2EP2-100ME

DEJAVA 보고서

2015004584

배지운

1. 서론

L4G2EP2-100ME(이하 L4G)를 진행하면서 많은 것들을 배울 수 있었지만 가장 기억에 남는 교훈은 역시 숙제는 미리 해두자 입니다. 이전에 다양한 생각과 계획들이 L4G 과제를 당일 착수하면서 많은 부분 구현되지 못했습니다. 아래에서 지금까지 완성된 플레이어(이하 DEJAVA)가 어떻게 작동되고 어떤 논리적인 구상이 반영되었으며 반영되지 못한 계획은 어떤 것이 있는지 이런 판단을 하게 된 근거는 어떤 것이 있는지 설명하도록 하겠습니다.

1. DEJAVA

먼저 DEJAVA를 이루는 모든 논리적 사고의 발상은 ‘잉크 방울’ 모델로부터 시작되었습니다. 분명 많은 데이터를 얻을 수 있다면 보다 정확한 판단을 할 수 있겠죠. 하지만 DEJAVA의 시야는 제한되어 있고 맵 전체의 시야를 얻기 위해서 영혼 상태를 여러 번 취하는 것은 점수의 손실로 반영되었습니다. 따라서 얻을 수 있는 모든 정보를 가지고 새로운 정보를 예측하는 모델을 만들었습니다. 어떤 플레이어가 (x, y)에 존재 했었다면 다음 턴의 그 플레이어는 확률적으로 갈 수 있는 모든 방향에 존재할 수 있을 것 입니다. 만약 생존자라면 {(x+1, y), (x-1, y), (x, y+1), (x, y-1)}에 존재할 수 있겠죠. 각각 모든 점에 0.25씩 더해서 확률적으로 바라볼 수 있을 것 입니다. 해당 부분은 *initialize()*의 *calculate expect variables*부분에 구현되어 있습니다. 이런 기초적인 틀을 가지고 매 턴에 사용해야할 의사 결정 상수들을 잘 측정해서 대입했습니다. 여기에 맵 가중치를 더해서 판단에 활용했습니다. 맵 가중치는 강의실의 어느 부분이 좀 더 불리한 가를 나타냅니다. 구석으로 몰릴수록 갈 수 있는 방향의 개수는 줄어들고 죽을 확률이 높기 때문에 맵 가중치를 e^(distance from center) 꼴로 나타내어 판단에 활용했습니다. 이 맵 가중치는 Mathematica를 활용하여 [시각화](https://github.com/MaybeS/ITE2037/blob/master/L4G2EP2/doc/report/res/CLASS_EXP_DFAULT.png)[[1]](#endnote-1) 하여 판단했습니다.

이런 기본적인 상수들과 기본 모델, 어떤 점수를 쫓는 것이 결과적으로 더 높은 학점을 얻을 수 있을 것인가에 대한 판단은 전적으로 실험을 통해서 이루어졌습니다. 주어진 L4G의 프로젝트 코드들을 몇 개 임의로 변경하여 각 실험의 모든 점수를 기록해 최종적인점수와 평균을 구해서 어떤 방향성을 갖는지 실험해 보았습니다. 본래 각 점수를 추구하는 컨셉 모델을 간단하게 만들어서 실험하려고 했으나 주어진 기본 봇의 성능이 월등해 그 봇 코드를 차용해 실험했습니다. 이 [변경](https://github.com/MaybeS/ITE2037/commit/6cfbeac6491e8b255fe5fb2b52670878b9c97aee)[[2]](#endnote-2)은 기본코드에서 어느 부분을 바꿨는지 가시적으로 보여줍니다. 이런 실험을 토대로 수 많은 변수들을 조작해가면서 어느 부분이 최적의 결과를 낼 수 있는지 판단했습니다. 우선 기본적인 강의실 상태를 5명의 혼돈 플레이어, 5명의 외톨이, 20명의 포식자, 70명의 시체 폭탄 코드가 추가된 정찰자를 기본으로 두고 DEJAVA를 추가해서 실험했습니다. 이 [로그](https://github.com/MaybeS/ITE2037/raw/master/L4G2EP2/doc/report/log.zip)[[3]](#endnote-3)는 각각 조건에서 어떤 결과가 나타났는지 보여줍니다. 이전 실험들을 통해서 기본 봇 모델 중 정찰자(시체 폭탄을 추가한)의 점수가 평균적으로 월등 했기 때문에 기본 모델로 정찰자를 채용하고 여기에 ‘잉크 방울’ 모델을 추가하기 시작했습니다. 또한 각종 상수 역시 여러 번의 실험을 통해서 높은 값을 찾는 방향으로 진행했습니다.

기본적으로 DEJAVA는 생존 점수와 시체 점수를 추구합니다. 우선 생존자의 상태에서는 생존 점수와 정찰 점수를 얻기 위해 노력합니다. 최우선적으로 감염자들이 적을 것으로 예측되는 곳을 향합니다. 하지만 정찰 점수를 얻기 위해 시야에 감염자가 있으면 추가 가중치를 적용해서 바로 옆에는 없지만 시야범위에는 감염자가 존재하도록 이동합니다. 이 이동에서도 데이터를 계속 갱신하여 신뢰성을 높이고 시야 범위에 없는 상태에선 좀 더 적은 가중치를 적용하여 좀 더 부정확한 판단임을 암시적으로 나타냈습니다. 감염상태에서는 빠르게 정화 기도를 올려 다음 부활을 도모합니다. 만약 해당 위치에 시체가 나타날 확률이 높다면 적은 쪽으로 이동합니다. 영혼 상태에서는 데이터를 누적하고, 생존 점수가 일정 수준 이상이라면 이제 시체 점수를 얻기 위해서 감염자에 다가가 빠르게 죽고, 감염자가 많은 쪽으로 부활하여 시체 점수를 올립니다.

추가로 구현하고 싶었던 것들 중에서는 다른 플레이어의 점수를 예측하는 모델도 추가해서 생존과 시체 점수를 고정으로 확보하는게 아니라 상황에 맞게 파악해서 가능성 높은 점수를 얻도록 하고 싶어서 간단한 모델을 만들었었는데 점수에 대한 판단이 계속 바뀌기 때문에 고루 점수를 모으게 되어 총점이 더 낮아지는 현상이 나타나 폐기했습니다. 또한 좀 더 상세한 ‘잉크 방울’ 모델을 만들기 위해서 머신 러닝을 적용해 이전 판의 자료를 활용하고 싶었으나 각 경기가 독립적이 되어야 하므로 넣지 못했습니다.

1. 맺음말

L4G를 진행하면서 객체 지향적 사고를 비롯하여 많은 새로운 것들을 배울 수 있었습니다. 물론 숙제를 미리미리 해두자 가 가장 중요한 교훈이었지만 아이디어를 구체화하고 적용하는 과정에서 여러 시행착오, 기대 밖의 결과를 보면서 어떻게 대응해야 하는지 좀 더 배울 수 있었고 매우 의미 있는 실습으로 기억에 남을 것 입니다. L4G를 하면서 가장 어려웠던 점은 아무래도 규칙이 너무 많은 점이 아닐까 싶습니다. 저는 기억력이 좋은 편이 아니라서 각 점수를 모두 기억하기 어려웠습니다. 또한 별개로 적용되는 규칙이 너무 많아서 각 조건을 구체화 시키는 과정 자체가 매우 힘들었습니다. 물론 L4G는 매우 흥미로운 게임이었고 DEJAVA를 구상하는 과정은 즐거웠지만 좀 더 적은 규칙으로 현재의 재미와 흥미로움을 유지하면 좋을 것 같습니다. 추구해야하는 점수 폭이 매우 넓어 선택의 폭을 키우고 과목당 점수를 계산하는 방식은 좀더 다양한 플레이를 가능하게하는 좋은 방식이었다고 생각합니다. 하지만 너무 다양한 플레이어가 존재할 수 있다 보니 상대방을 예측하는 과정은 꽤 힘들었습니다. 제가 예측했던 포식자3: 정찰병7도 풍문에 의하면 완전히 틀린 예측이었다고 하며 플레이 역시 완전히 예측하기 힘들었습니다. 저는 모든 사람들이 점수를 높게 받기 위해서 최선의 선택을 한다고 고려했는데 실상은 아니었고 정말 여러 종류의 플레이어를 구상한 사람들을 만나볼 수 있었습니다. 여기서 포인트는 우리가 다양한 플레이어에 대응하는 각각의 모델을 구상할 것이냐 아니면 어떤 하나의 점수를 집중으로 공략해서 많은 점수를 얻을 것이냐로 갈릴 수 있습니다. 총체적으로 모든 요소가 운에 결정되는 것은 아니지만 결국 상대가 어떤 전략을 세울 지에 관한 어떠한 정보가 없기 때문에 판단에 오류가 있을 수 있고 이게 결과적으로 점수에 반영되기 때문에 상대방의 허를 찌르는 작전을 세워야 하는데 사람수가 너무 많다 보니 그러한 예측이 보기 좋게 빗나갔다고 생각해볼 수 있다.



1. https://github.com/MaybeS/ITE2037/blob/master/L4G2EP2/doc/report/res/CLASS\_EXP\_DFAULT.png [↑](#endnote-ref-1)
2. https://github.com/MaybeS/ITE2037/commit/6cfbeac6491e8b255fe5fb2b52670878b9c97aee [↑](#endnote-ref-2)
3. https://github.com/MaybeS/ITE2037/raw/master/L4G2EP2/doc/report/log.zip [↑](#endnote-ref-3)