안녕하세요 Transfer Matrix Method 발표를 하게 된 서성민입니다.

이번 주 계획은 완료하였으며 앞으로의 계획은 다음과 같습니다

지난 2차 발표 때 소개에서 수정한 사항으로 다음 함수가 더 적합하다 생각하여 수정하였습니다.

현재 진행 상황은 파일작성은 완료하였고 그래프를 출력해 봤습니다. 하지마 아직 Essential Macleod는 사용해보지 않았으며 Power가 음수로 나오는 때가 있는 등 문제점들이 남아 있습니다.(굴절률을 통해 세타값을 구하는 과정에서 투과가 없는 경우 등 예외상황에 따른 것으로 추측)

하지만 2차 발표에서 미달 사유였던 파장대를 통일하는 것은 이분탐색을 통한 선형근사로 해결 해 주었습니다. 중간고사 관계로 계획했던 Poynting Vector에 대한 이론적 소개는 간략화하고 (전자기학2에서 학습한 것과 동일함) 알고리즘 위주와 RGB컨버터에 대해 잠깐 소개해드리려 합니다.

알고리즘 개괄은 다음과 같습니다. 처음 층의 깊이를 입력해주면 정보를 받는 인터페이스를 출력하여 진행합니다. 현재 입사할 때와 출력은 Air로 설정해 놓았습니다.

그래프는 아직 모델 설정을 하지 않아 1개의 매질만 지나는 간단한 것으로 그려 봤습니다. 빛으로 임의의 LED등의 Spectral Power Distribution을 썼습니다. (a)와 같이 이름을 입력하면 상수가 저장된 폴더를 찾아가 값들을 입력 받습니다. (b)는 그래프로 나타낸 것으로 다음과 분포 및 갈색으로 나타날 것이라는 결과를 보여줍니다. (c)는 Glass의 파장별 굴절률을 (&k) 그리고 (d)는 빛의 파장별 분포를 csv파일로 저장하고 있습니다. (e)를 통해 파장대를 390(nm)-830(nm), 0.1(nm)간격으로 바꿔줍니다. (f)는 파장대별 wave\_vector의 상수부분을 저장합니다. (g)는 반복 구문 속에 들어가 있어서 층마다 반복하며 굴절각과 S-pol, P-pol의 경계에서 투과 반사를 계산하여 다음 층에 자신의 굴절률 및 입사각을 제공해 줍니다. 굴절각은 (h)와 같이 구했습니다. 이를 토대로 총 투과율을 구하고 이를 빛에 적용하는 것을 (i)와 같이 구현했습니다(세타도 허수 가능). 다음 그림은 S-pol 그리고 P-pol에서 Poynting Vector 그리고 이를 통해 Power의 투과율을 계산하는 식으로 마지막 층에서 들어오는 빛이 없다고 할 때 E(backward)에 0을 넣어주면 자연스럽게 나옵니다.

이를 (j)와 같이 구현했으며 P,S의 파워를 다시 합쳐서 돌려 줬습니다. 다음으로 (k)는 색좌표계에서 rgb와 xyz 사이의 관계를 나타내주는 행렬입니다. 제가 xyz 실험값을 가지고 있으므로 이를 통해서 rgb 값으로 수정해줍니다. (j)를 통해 받은 값을 (l)에 적용하여 색을 구합니다. (m)과 (n)은 dot product를 통해서 위에 말한 과정을 처리하는 것입니다. (K)는 (o)에서 상수행렬을 반대편으로 옮겨준 것이며 (p)의 인테그럴은 서메이션으로 (0.1nm steps) 대체 했습니다. 마지막으로 (q)(r)은 처음 말씀 드린 Power가 음수가 나오는 경우로 색을 구하고자 하면 당연히 음수에 의한 에러가 나옵니다.

개인적으로 450nm 대의 파장은 투과가 있으며 600nm 근처의 파장은 반사시키기에 코드에 이를 고려하지 못해 다음과 같은 잘못된 출력이 나오는 것이라 생각합니다. 이에 대해서는 다음 발표에서 Essential Macleod를 통한 확인 및 확실히 설명할 수 있도록 하겠습니다.

이상으로 발표를 마칩니다.