|  |  |  |
| --- | --- | --- |
| **[EEC3600-001] 수치해석** | | |
| 소속: 전기전자공학부 | 학번: 12191529 | 이름: 장준영 |
| **Term Project** | | **Prob #1** |

1. **Problem**
2. 문제

|  |
| --- |
|  |
|  |
|  |

1. **Solution (a)**

|  |
| --- |
|  |
| ■ **Case 1: Fastest Settling Time (최소 시간 도달)**  • 은 각 시간까지의 가속도 누적합이 속도를 형성함을 나타내는 Lower Triangle Matrix이다.  • 은 각 시간에서 도달해야 하는 목표 변화량으로 구성되며, 모두 같은 값으로 채워진다.  • 최소제곱 문제 를 Normal Equation 를 통해 해석적으로 계산한다.  • 은 초기 속도에 의 누적합을 더하여 시간에 따른 속도 변화를 계산한 것이다.​  ■ **Case 2: Minimum Control Signal (제어 입력 최소화)**  • 는 상단은 항등 행렬(각 최소화를 유도), 마지막 행은 조건을 표현.  • 는 상단은 0, 하단은 총 변화량: 제어 입력 크기는 최소화하되, 총합은 고정된다.  • 해는 제어 입력이 균등하게 분산된 형태로 나오며, 값이 작고 일정하다.  • 는 누적합을 통해 속도를 계산한다.​  ■ **Case 3: Trade-off Between Speed and Control Effect (트레이드오프)**  • 은 상단은 속도 조건(Case 1), 하단은 제어 입력 항에 대한 패널티를 반영하였다.  • 는 제어 입력 크기를 얼마나 강하게 패널티를 줄지 결정하는 하이퍼파라미터이다.  • 는 상단은 목표 속도 변화량, 하단은 제로 패널티이다.  • 결과적으로, 제어 입력이 초기에는 크고 점차 줄어드는 현실적인 곡선이 도출된다.​ |
|  |

1. **Solution (b)**
2. **문제 분석**

본 문제는 선형 이산 시스템 하에서, 초기 속도 에서 목표 속도 까지 도달하도록 제어 입력 를 구하는 최적 제어 문제이다. 목적함수의 설정에 따라 서로 다른 해를 갖게 되며, 이는 현실적인 제어 시나리오를 수치적으로 시뮬레이션하고 평가하는 데 매우 적합하다.

1. **해(Solution) 분석**

이번 과제에서 사용한 해법은 모두 선형 최소제곱 문제의 Normal Equation 해법인

를 기반으로 하였다. Python의 numpy 패키지를 통해 간단하게 구현 가능하며, 차원이 작을 경우 수치적 안정성에도 문제가 없다.

* **Case 1 해의 구조**
* 해는 첫 번째 성분 만 크고, 나머지는 0인 형태로 주어진다.
* 이는 가 Lower Triangle Matrix이고, 가 일정한 상수 벡터이기 때문에 발생하는 구조적 특성이다.
* 즉, 모든 속도 오차를 단 한 번의 제어로 상쇄하려고 하는 해이다.
* **Case 2 해의 구조**
* 의 상단이 항등 행렬이기 때문에, 해는 각각의 가 최소가 되는 방향으로 분산된다.
* 마지막 행이 총합 제약을 추가함으로써, 모든 가 일정하게 나오는 균등분산형 해가 도출된다.
* 특히 이 해는 가 일종의 “평균값”처럼 분포되는 형태로 해석할 수 있다.
* **Case 3 해의 구조**
* 는 속도 누적항과 제어 입력 패널티 항을 함께 고려하므로, 두 제약이 동시에 작용한다.
* 그 결과, 는 크고 이후 는 점차 감소하는 형태의 비선형적인 감소 곡선이 도출된다.
* 값이 클수록 제어 입력이 더 억제되어, Case 2에 가까운 결과가 나오고, 작을수록 Case 1과 유사해진다.
* 이 해는 실제 제어 시스템에서 매우 자주 사용되는 정상상태 접근 전력과 매우 유사하다.
* **수치적 안정성과 구현 관점**
* 세 가지 해 모두 가 양의 정부호 행렬이므로 해는 유일하고 안정적으로 존재한다.
* 단, 문제 크기가 커질 경우 의 condition number가 커질 수 있어, normalization 또는 SVD 기반 해법이 필요할 수 있다.