Check for updates

Journal of Institute of Control, Robotics and Systems (2025) 31(2):78-86
https://doi.org/10.5302/J.ICROS.2025.24.0253 ISSN:1976-5622 eISSN:2233-4335

가우시안 프로세스 회귀 모델을 활용한 불확실성이 있는 시스템의 예측 제어 튜토리얼

Tutorial on Predictive Control for Systems With Model Uncertainty Using Gaussian Process Regression

김 정 효, 박 수 용, 한 경 석*
(Junghyo Kim[1], Suyong Park[1], and Kyoungseok Han[1,*])
[1]Department of Automotive Engineering, Hanyang University

Abstract: This study presents a method to improve control performance in complex dynamic systems by integrating Gaussian processes (GP) and model predictive control (MPC). GP enables learning from nonlinear systems without explicit models, while MPC can integrate the learned models to generate optimal control policies. Approximation techniques such as sparse GP, which focus on computational efficiency, are introduced to address the high computational load of GP. Furthermore, experimental and comparative analyses consistently show that GP-based MPC surpasses traditional nominal model-based methods in controlling nonlinear systems, especially in terms of tracking accuracy, robustness to uncertainties, and effective handling of unmodeled dynamics. This article highlights the theoretical background, computational considerations, and practical applications of GP–MPC, providing a foundation for exploration of learning-based control systems.

Keywords: gaussian process regression, model predictive control, learning-based control, model uncertainty

I. 서론

모델 예측 제어(MPC)는 시스템 모델을 활용, 특정 시간 동안 미래의 시스템 상태를 예측하고 최적의 제어 입력 결정을 통해 시스템의 미래 상태를 최적화하는 제어 기법으로, 최근 상용화 연구가 활발히 진행되고 있는 편이다[1,2]. 특히, MPC는 여러 개의 입력과 출력을 동시에 다룰 수 있어 복잡한 시스템 제어에 유리하고, 화학 공정 제어, 자동차 산업, 항공우주 시스템 등에 널리 적용되는 추세로, 높은 제어 성능과 안정성을 제공하는 강력한 제어 기법으로 인정받고 있다.

하지만, MPC는 시스템 모델을 기반으로 미래의 시스템 동작을 예측하기 때문에 모델이 부정확할 경우 제어 성능이 저하되는 단점을 가지는데, 부정확한 모델로 제어기가 설계될 경우, 잘못된 시스템 동작 예측에 기반한 제어 입력이 도출되고, 이는 시스템의 안정성과 같은 중요한 문제로 이어질 수 있다. 또한, MPC는 각 샘플링 단계마다 최적화 문제를 해결해야 하므로 계산 부하로 인한 실시간 제어가 필요한 임베디드 시스템 적용에 어려움도 있다.

지난 수십 년간 MPC의 이러한 단점을 해결하기 위해 다양한 접근법이 연구되었다. Tube-based stochastic MPC는 시스템의 불확실성을 튜브 형태로 표현하여, 실제 시스템이 이 튜브 내에서 동작하도록 제어함으로써 모델의 오차에 대한 내성을 강화한다[3]. 예측된 상태 경로를 중심으로 불확실성을 감안한 허용 오차 범위를 설정하며, 이를 통해 모델 오차나 외란에 의한 상태 변동을 효과적으로 억제할 수 있다. 또한, stochastic MPC는 시스템의 불확실성을 확률적 모델로 고려하여 제어 입력을 설계함으로써 모델의 불확실성에 대한 견고성을 향상시킨다[4]. 이 방법은 시스템 동작의 확률 분포를 예측하고, 이러한 불확실성을 반영하여 최적의 제어 전략을 도출한다. 예를 들어 노이즈나 외란이 존재하는 환경에서도 안정적인 제어를 보장할 수 있다. Chance-constrained MPC는 시스템의 상태가 특정 제약 조건을 만족할 확률을 보장하도록 제어기를 설계하는 방법이다[5]. 이러한 접근법은 불확실한 환경에서도 시스템이 안전한 상태를 유지할 수 있도록 도와준다.

한편, MPC의 계산 부담을 경감하기 위해 연구자들은 최적화 문제의 구조적 특성을 활용한 수치적으로 효율적인 솔버 개발에 집중해왔다. 예를 들어 QP (Quadratic Programming) 문제의 특성을 이용한 고속 솔버나 분산 최적화 기법이 적용되고 있다[6,7,8].

최근에는 이러한 MPC의 단점을 머신러닝(ML)과 통합한 학습 기반 제어 방법이 활발히 연구되고 있다. ML은 MPC 제어 문제에서 시스템 모델 학습, 계산 비용 감소, 강화 학습 결합, 안전 필터 적용 등의 방면으로 적용될 수 있다. [9]에서는 서포트 벡터 머신(SVM)을 활용하여 MPC 모델의 정확도와 성능을 높이는 방법을 보여주었다. [10,11]에서는

* Corresponding Author
Manuscript received November 2, 2024; revised December 4, 2024; accepted January 11, 2025
김정효: 한양대학교 미래자동차공학과 대학원생(hyo05122@hanyang.ac.kr, ORCID 0009-0009-5775-0252)
박수용: 한양대학교 미래자동차공학과 대학원생(suyongpark@hanyang.ac.kr, ORCID 0009-0009-5281-7665)
한경석: 한양대학교 미래자동차공학과 부교수(kyoungsh@hanyang.ac.kr, ORCID 0000-0002-4986-2053)
※ 본 연구는 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 연구되었음(No. NRF-2021R1C1C1003464).

Copyright© ICROS 2025

MPC의 제어 동작을 뉴럴 네트워크(NN)을 통해 근사하여 실시간 계산 시간을 줄이는 접근법을 소개한다. 이 방법은 최적화된 제어 입력을 근사함으로써 실시간 제어 구현을 가능함을 보였다. [12]에서는 MPC를 강화학습(RL)과 같은 학습 기반 제어기의 안전 필터로 사용하는 방법을 다룬다. 이 접근법은 MPC를 기반으로 하여 상태 및 입력 의존 불확실성을 고려하며, RL 기법을 안전하게 해결하기 위한 시도이다. [13,14]에서는 ML을 MPC 최적화 계산에 적용하여 계산 비용이 높은 연산에서 시간을 줄이는 방법을 설명한다. K-NN (K-Nearest Neighbor) 분류기나 HMM (Hidden Markov Model)과 같은 ML기법을 활용하여 active set method 기반 최적화 문제의 반복횟수를 줄여 수렴의 시간을 단축할 수 있음을 보였다.

특히, 복잡한 비선형 시스템이나 파라미터 불확실성이 있는 시스템 모델을 학습하여 MPC의 모델 의존성 문제를 효과적으로 해결하기 위한 접근법에는 뉴럴 네트워크, SVM, 가우시안 프로세스(GP) 등의 기법들을 적용한 연구들이 수행된 바 있다. [15,16]에서는 시스템의 입력과 출력 데이터로 NN을 학습하여 시스템 모델을 식별하는 방법을 제안하였다. ML 기법 중 하나인 auto-encoders와 뉴럴 네트워크를 기반으로 비선형 모델의 출력 및 상태 업데이트를 식별하고 모델 적합 능력을 보였다. [17]에서는 시스템의 비선형적 거동을 LPV (Linearly Parameter-Varying) 모델로 나타 내기 위해 최소 자승법 기반의 SVM을 적용하여 데이터 기반의 상태 공간을 모델링하였다. 학습 기반 제어 방법 중에서 가우시안 프로세스(GP)를 활용하여 모델 정확도를 개선하고, 개선된 모델을 MPC에 적용하여 모델 불확실성을 해소한 GP-based MPC (GP–MPC)도 주목을 받고 있다[18].

GP는 확률론적 접근 방식을 통해 데이터의 불확실성을 정량화하고, 예측 결과에 대한 신뢰도를 제공할 수 있는 점에서 뉴럴 네트워크와 차별화된다. NN은 모델의 불확실성을 직접적으로 다루지 못한다는 한계가 있으며, 반면 GP는 적은 데이터로도 우수한 성능을 발휘하면서 모델의 불확실성을 확률적으로 표현할 수 있다. 따라서, GP를 통한 불확실성 정량화는 예측 오류를 줄이고 시스템의 안전성을 향상시키며, MPC의 모델 정확도 문제를 해결하는 데 핵심적인 역할을 수행할 수 있다[19,20].

다만, 대규모 데이터나 빠른 샘플링 속도가 요구될 때는, GP의 계산 복잡도가 커서 실시간 제어 응용에 어려움이 따른다. 이를 해소하기 위해, SOD (Subset of Data), FITC (Fully Independent Training Conditional), VFE (Variational Free Energy) 등과 같은 sparse GP 기법들이 제안되어 왔다[21-23]. 이 기법들은 유도점(inducing points)만을 추려서 전체 공분산 행렬을 근사하는 방식으로, 모델 정확도와 계산 효율성 사이의 적절한 균형을 찾는다. 예를 들어, FITC는 유도점 위치를 최적화해 로그 우도를 최대화함으로써 예측 정확도를 유지하고, VFE는 변분추론을 통해 데이터 적합도와 근사 분포의 복잡도 사이의 균형을 찾는 지표로 evidence lower bound (ELBO)라는 목적함수를 최대화함으로써, GP의 계산 효율성과 근사 정확도를 향상시킨다. 또한, 이러한 기법들에서 계산 효율성을 유지하면서 합리적인 정확도를 유지하는 유도 포인트의 수 결정이 필요하다.

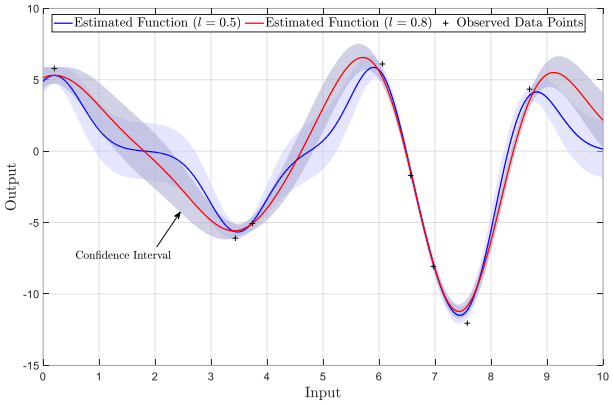

그림 1. 서로 다른 길이 척도에 대한 가우시안 프로세스 회귀와 신뢰구간.

Fig. 1. Gaussian process regression with different length-scale parameters and confidence intervals.

GP의 확률론적 모델과 MPC의 미래 예측 및 최적화 기능을 결합한 GP–MPC는 기존 MPC의 단점을 보완하여 모델링 오차와 비선형성을 효과적으로 처리하고 예측 정확도를 높일 수 있다[24,25]. 특히, GP–MPC는 데이터 기반 모델링을 통해 시스템의 불확실성을 정량화할 수 있기 때문에 안정적인 제어가 가능하다. [26]에서는 차량의 adaptive cruise control (ACC) 시스템에서 차량의 가속도 모델에 외란을 GP 통해 학습하여 제어 성능을 향상시켰다. [21]에서는 오프라인으로 선박의 거동 데이터를 수집하여 GP기반의 상태 공간 방정식을 통해 MPC 문제를 해결하였다. [24,27]에서는 오프라인으로 데이터 샘플을 생성하고, 실제 제어 환경의 상태 변화를 반영하기 위해 새로운 관측 데이터를 통해 GP 학습 모델을 온라인으로 업데이트하여 MPC와 통합하였다.

본 논문에서는 이러한 선행 연구들을 바탕으로 GP와 MPC를 통합하는 이론적 배경을 설명하고, 오프라인 GP 모델 수립 및 MPC에 활용 시 발생하는 계산 복잡성을 해결하기 위한 방법에 대해 소개한다. 특히, GP의 확률론적 특성을 통해 예측 오류를 줄이고, 복잡한 비선형 시스템에서도 데이터 기반의 시스템 모델을 학습할 수 있음을 시뮬레이션을 통해 검증하고자 한다.

본 논문의 구성은 2장에서 GP를 통한 시스템 모델링 방법에 대해 정의하고, 3장에서 GP와 MPC간의 통합 방법을 소개한다. 4장에서는 응용 시뮬레이션을 통하여 일반 MPC와 GP–MPC 간의 성능을 비교 분석하여 모델 정확도에 따른 제어 성능의 차이를 분석한다. 마지막 장에서는 GP–MPC가 실제 제어 시스템에 어떻게 적용될 수 있는지 논의하며, 향후 학습 기반 제어 시스템의 발전 가능성과 응용 방향을 제시한다.

II. 가우시안 프로세스

GP는 베이지안 기계 학습의 대표적인 모델로, 불확실성을 가진 시스템의 동작을 예측하기 위한 목적 함수를 근사화하는 확률적 프레임워크를 제공한다. GP는 평균 함수와 공분산 함수로 정의되며, 이를 통해 확률적이고 비모수적인 (non-parametric) 모델을 구현하는 데 적합하다. 비모수적 모델의

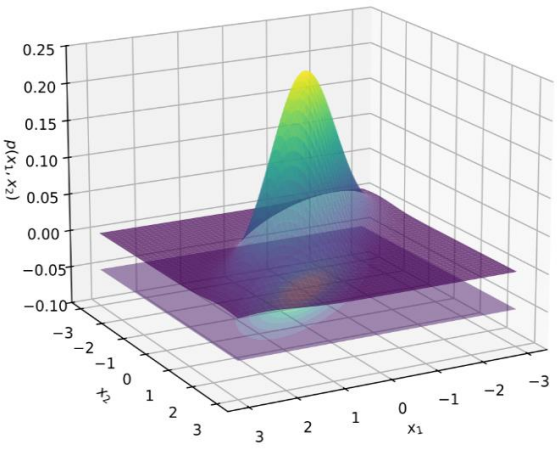

그림 2. 2차원 입력에 대한 RBF 커널 함수.
Fig. 2. RBF kernel function for 2-dimensional input.

특성은 데이터의 복잡한 패턴을 모델링하는 데 있어 유연성을 제공하고, 명시적인 모델 구조 없이 데이터로부터 직접적인 함수 근사가 가능하다[28].

결정론적인 모수적(parametric method) 모델과 달리, GP는 관찰된 데이터에 맞춰 이를 잘 표현할 수 있는 함수들에 대해 확률 분포를 추정하고, 관측된 데이터를 바탕으로 미지의 영역에 대한 데이터의 평균에 대한 예측과 함께 불확실성(분산)을 제공한다. 그림 1은 8개의 관측 데이터를 바탕으로 서로 다른 하이퍼파라미터를 가진 두 모델의 예측 결과를 보여준다. 관측된 데이터를 기반으로 평균 함수와 신뢰 구간을 계산하며, 공분산을 통해 불확실성의 크기를 정량화할 수 있다. 데이터가 밀집된 구간에서는 공분산이 작아 불확실성이 낮아지고, 반대로 데이터가 희소한 구간에서는 공분산이 커져 불확실성이 증가하는 것을 알 수 있다. 이러한 가우시안 프로세스의 특성은 다음 절에서 평균 함수와 공분산 함수를 통해 다룬다.

1. 가우시안 프로세스 회귀

가우시안 프로세스 회귀(GPR)는 입력 데이터 Z에 해당하는 관측값 Y의 데이터 집합 $D = \{Z, Y\}$ 을 통해 시스템에 대한 평균 함수 및 불확실성을 나타내는 공분산을 모델링하는 비모수적인 방법이다. 사전에 데이터 분포를 나타내는 함수 형태를 지정할 필요 없이 입력 데이터 벡터 ($Z = [z_1, \dots, z_n]$) 에 해당하는 출력 값 ($f = [f(z_1), \dots, f(z_n)]$) 을 분석하여 함수의 분포를 모델링한다. GPR에서는 관측 데이터의 분포를 다변량 정규분포(multivariate normal distribution)으로 설명할 수 있다고 가정하며, 평균 함수 $m(Z)$와 공분산 $K(Z, Z')$로 모델링한다. 사전 분포(prior distribution)는 다음과 같은 형태로 정의된다.

$$f(Z) \sim \mathcal{GP}\big(m(Z), K(Z, Z')\big) \tag{1}$$

$$m(Z) = E[f(Z)] \tag{2}$$

$$K(Z, Z') = E\big[\big(f(Z) - m(Z)\big)\big(f(Z') - m(Z')\big)\big] \tag{3}$$

식 (1)과 같이, 사전분포 모델 $f(Z)$ 는 가우시안 프로세스를 따른다고 가정하고, 이를 평균 함수 $m(Z)$ 와 커널 함수를 기반으로 만든 공분산 함수 $K(Z, Z')$를 통해 나타낼 수 있다. 식 (2)의 평균함수 $m(Z)$ 는 각 입력 Z에서 함수 $f(Z)$의 기대값을 나타낸다. 식 (3)의 공분산 함수는 두 입력 Z 와 Z' 간 의 함수 값의 상관관계를 나타내어 입력 데이터 간의 유사도를 정량화하는 공분산 값으로, 입력 데이터 간의 관계를 학습함으로써 새로운 입력 값에 대한 예측의 정밀도를 높인다.

공분산 함수의 선택은 GPR 성능에 중요한 영향을 미치며, 주로 RBF (Radial Basis Function) 및 Matern 커널 등이 주로 사용된다[29]. 특히, 커널 함수는 모델의 성능을 조절하는 하이퍼파라미터를 포함하고 있기 때문에, 주어진 데이터에 적합한 최적의 하이퍼파라미터를 선택하는 것이 중요하다.

이와 같이 사전 분포는 훈련 데이터를 학습하기 전에 함수의 가능한 분포를 나타내며, 평균 함수값과 커널 함수의 하이퍼파라미터에 의해 결정된다. 훈련 단계에서는 관측된 데이터 $D = \{(z_i, y_i)\}_{\{i=1\}}^{n}$ 을 통해 하이퍼파라미터를 최적화한 후, 이를 바탕으로 사후 분포(posterior distribution)를 업데이트하여 예측 모델을 생성한다.

하이퍼파라미터 최적화는 관측된 데이터로부터 주어진 모델의 가능도를 최대화하는 하이퍼파라미터를 찾는 방법인 최대 우도법을 통해 수행되며 다음과 같이 표현할 수 있다.

$$\Theta^* = \arg\max_{\Theta} \log p\,(Y \mid Z, \Theta) \tag{4}$$

식 (4)의 $p(Y \mid Z, \Theta)$는 주어진 입력 Z와 하이퍼파라미터 Θ에서 관측값 Y가 발생할 확률인 우도함수를 나타낸다.

사전 분포 모델만으로는 새로운 데이터 관측에 따른 평균 및 공분산을 예측하기 어렵다. 이를 해결하기 위해 베이지안 관점에서 기존 데이터 Z와 새로운 입력 값 z_*를 결합하여 가우시안 프로세스의 사후 분포를 구하게 된다. 사후 분포 모델은 주어진 훈련 데이터와 새로운 입력 값 z_*에서의 함수 값의 불확실성을 고려하며 평균 μ_*와 공분산 Σ_*로 다음과 같이 표현된다.

$$\begin{bmatrix} f_* \\ f \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_*(z_*) \\ \mu(z) \end{bmatrix}, \begin{bmatrix} K(z_*, z_*) & K(z_*, Z) \\ K(Z, z_*) & K(Z, Z) \end{bmatrix}\right) \tag{5}$$

$$\mu_* = m(z) + K(z_*, Z)(K(Z, Z) + \sigma_n^2 I)^{-1}\big(Y - m(Z)\big) \tag{6}$$

$$\Sigma_* = K(z_*, z_*) - K(z_*, Z)(K(Z, Z) + \sigma_n^2 I)^{-1} K(Z, z_*) \tag{7}$$

식 (5)는 기존 데이터 Z를 통해 새로운 입력 값 z_*의 결합 분포를 나타내며 베이즈 정리를 사용해 식 (6), (7)와 같이 사후 분포를 계산한다. 식 (6)에서 μ_*는 새로운 입력 z_*에 대한 사후 평균으로, 기존 데이터와의 공분산 $K(z_*, Z)$ 와 관측값 Y를 사용하여 계산된다. 이때 $K(Z, Z)$는 기존 데이터 Z에서의 공분산 행렬이며, $\sigma_n^2 I$는 관측 노이즈를 반영한다. 식 (7)과 같이 예측된 분산 Σ_*은 관측값 Y가 아니라 입력 지점들 간의 관계에 따라 결정된다. 입력 지점들의 위치에 따라 예측의 불확실성이 결정된다는 점에서, GP는 입출력의 복잡한 관계를 유연하게 모델링하고 불확실성을 예측할 수 있다[30].

그림 2는 RBF 커널 함수의 형태를 시각화한 것으로 입력 변수 x_1과 x_2사이의 거리에 따른 유사도를 나타내며 거리가 멀어짐에 따라 커널 함수의 값이 감소하는 결과를 보여준다. 이와 같이 커널 함수는 입력 공간에서 가까운 데이터 포인트

들이 유사한 출력을 가질 가능성이 높다는 가정을 기반으로 모델의 정확한 회귀 결과를 얻는 데 중요한 역할을 한다. n 개의 훈련 데이터에 대해 공분산 함수는 식 (8)과 같이 입력 포인트 z 간의 커널로 구성된다.

$$K(Z,Z)=\begin{bmatrix} k(z_1,z_1) & \cdots & k(z_1,z_n) \\ \vdots & \ddots & \vdots \\ k(z_n,z_1) & \cdots & k(z_n,z_n) \end{bmatrix} \quad (8)$$

실제 문제에서는 대부분의 관측값은 노이즈를 포함하므로 가우시안 노이즈의 분산 σ_n 을 포함한 $K(Z,Z)+\sigma_n^2 I$ 의 역행렬을 계산한다. 본 논문에서는 커널 함수 $k(z,z^*)$ 로 일반적으로 널리 사용되는 함수 중 하나인 지수 제곱 커널 함수를 사용하고, 이는 식 (9)와 같이 정의된다.

$$k(z,z')=\sigma_f^2\, exp\left(-\frac{1}{2}(z-z')^{\top}M^{-1}(z-z')\right) \quad (9)$$

식 (9)에서 σ_f 는 신호 분산 스케일(signal variance)로 예측 범위를 조절하며, $M=diag(l_1,\dots l_d)$ 은 양의 대각 행렬이며 d 는 z 의 차원의 크기로 다차원 입력에 따른 각 차원 영향도를 조절한다. 연속성, 매끄러움, 주기성 등 특정 데이터 속성에 따라 커널의 선택지가 존재하며, 이에 따른 하이퍼 파라미터의 구성과 최적화가 필요하다.

2. Sparse GP

식 (6)과 (7)의 사후 평균 및 공분산 계산에서 요구되는 계산량은 훈련 데이터 수 n에 따라 각각 $O(n^2)$ 과 $O(n^3)$ 으로 증가하기 때문에 대규모 데이터셋에서 큰 계산 부하를 유발한다. 특히, 데이터의 수가 많을수록 역행렬 계산으로 인한 연산 부하는 혜를 실시간 제어 문제에 활용하는데 걸림돌이 된다고 할 수 있다.

이러한 연산 부하 문제 해결하기 위해 sparse GP 기법이 고려될 수 있다. Sparse GP은 전체 데이터셋이 아닌 일부 샘플 데이터를 사용하여 모델을 구성하는데, 유도점을 사용하여 GP 모델을 근사한다[31]. Sparse GP은 전체 데이터셋의 정보를 축소된 유도 포인트 집합 $U=[u_{1,ind},\dots,u_{m,ind}]$) 으로 압축하는 데 초점을 맞추고 있으며, 공분산 행렬의 역행렬 계산 시 필요한 크기를 줄여 계산 복잡도를 낮춘다. 따라서 역행렬을 계산해야 하는 공분산 행렬의 크기를 줄여서, 계산 복잡도를 전체 훈련 데이터 포인트 수 n 이 아닌 유도 포인트 m의 수에 의존하도록 줄일 수 있다. Sparse GP 모델의 계산 복잡도는 평균 계산 시 $O(m^2)$, 분산 계산 시 $O(m^3)$ 으로 축소되어, 기존 GP에 비해 실시간 제어 문제에 용이하게 적용될 수 있다.

$$K(Z,Z)\approx Q(Z,Z)=K(Z,U)K(U,U)^{-1}K(U,Z) \quad (10)$$

$$\boldsymbol{\Lambda}=\mathrm{diag}(K(Z,Z)-Q(Z,Z)+\sigma_n^2 I) \quad (11)$$

식 (10)에서 $K(U,U)$는 유도 포인트 간의 공분산 행렬이며 기존 공분산 $K(Z,Z)$ 을 근사화한 행렬이다. 식 (11)의 $\boldsymbol{\Lambda}$ 는 실제 공분산과 근사된 공분산 간의 차이를 계산한 후, 노이즈를 더한 대각 행렬이다. 근사된 공분산을 통한 입력 벡터 z에 대한 예측 평균과 분산은 다음과 같다.

$$\mu_z=m(z)+Q(z,Z)(Q(Z,Z)+\boldsymbol{\Lambda})^{-1}\big(Y-m(Z)\big) \quad (12)$$

$$\Sigma_z=K(z,z)-Q(z,Z)(Q(Z,Z)+\boldsymbol{\Lambda})^{-1}Q(Z,z) \quad (13)$$

이러한 sparse GP에서는 훈련 데이터로부터 유도 포인트를 적절히 선택하는 것이 모델의 정확성과 효율성을 유지하는 데 매우 중요하다. 이러한 포인트들은 데이터셋의 분포 특성을 반영하도록 선택되며, 정보 손실을 최소화하는 방식으로 데이터의 구조를 반영해야 한다[22]. 본 논문에서는 FITC 기법을 사용해 오프라인에서 유도점 최적화 수행하여 sparse GP를 적용하였다[32].

III. 가우시안 프로세스 기반 모델 예측 제어

3장에서는 2장에서 소개된 가우시안 프로세스와 모델예측 제어 기법 간의 통합을 통하여 데이터 기반 모델 예측 제어 프레임워크를 소개한다.

1. 가우시안 프로세스 기반 MPC

비선형 동적 시스템의 제어를 고려할 때, 이산 시간에서의 상태 방정식은 식 (14)과 같이 나타낼 수 있다.

$$x_{k+1}=f(x_k,u_k)+g(x_k,u_k) \quad (14)$$

여기서 $x_k\in\mathbb{R}^{n_x}$ 는 시스템의 상태 변수이며 $u_k\in\mathbb{R}^{n_u}$ 는 시간 k에서의 제어 입력이다. 상태 방정식은 불확실성이 고려되지 않은 정규 모델 $f(x_k,u_k)$ 와 불확실성 모델 $g(x_k,u_k)$로 구성된다. 데이터 수집을 통해 불확실성 모델을 가우시안 프로세스로 모델링하는 것이 핵심이다.

$g(x_k,u_k)$가 고려되지 않거나 부정확하게 모델링될 경우, MPC의 예측 성능을 저하시킬 수 있는 요인으로 작용하여 제어 성능 저하에 직접적인 영향을 미친다. 가우시안 프로세스를 통해 $g(x_k,u_k)$를 모델링하여 비선형 상태 방정식을 다음과 같이 표현할 수 있다.

$$x_{k+1}=f(x_k,u_k)+B_d(d(x_k,u_k)+w_k) \quad (15)$$

학습 대상인 g는 식 (15) 에서와 같이 GP 모델로 근사할 수 있다. 학습된 GP 모델은 $d(x_k,u_k)$ 와 노이즈 $w_k\sim N(0,\Sigma^w)$를 포함하고 있다. B_d 행렬은 학습 기반 모델의 영향을 특정 상태로 제한하여 계산 효율성을 높이는 데 사용한다. $B_d=\mathbf{I}$로 설정하면, 학습 기반 모델 $d(x_k,u_k)$가 모든 상태 변수에 영향을 미치도록 할 수 있다[17]. 학습된 GP 모델은 다음 식 (16)과 같이 표현할 수 있다.

$$d(x_k,u_k)\sim\mathcal{N}\left(\mu^d(x_k,u_k),\Sigma^d(x_k,u_k)\right) \quad (16)$$

$\mu^d(x_k,u_k)$는 평균이며, $\Sigma^d(x_k,u_k)$는 공분산이다. GP-MPC를 구현하기 위한 전체 구조는 그림 3이 보여준다. 정규 모델 기반 MPC로 시스템을 제어하여 학습을 위한 오차 데이터를 수집한다. 훈련데이터는 k시점에서 정규 모델이 예측한 $f(x_k,u_k)$상태와 시스템의 실제 상태 x_{k+1}의 차이를 통해 residual dynamics를 계산한다. 이렇게 수집된 오차 데이터 $\Delta x_{k+1}=x_{k+1}-f(x_k,u_k)$는 훈련 출력 데이터 Y로 사용되며, 이때 해당하는 시스템의 상태 x_k와 입력 u_k을 훈련 입력 데이터 Z로 하여 오프라인으로 학습한다. 이때 모델 정확도를 위해 상태 변수에 포함되지 않는 시스템

정보도 함께 입력 데이터로 필요 시 선택한다. 본 논문에서는 멀티 출력 GP모델 대신 단순성을 우선하여, 단일 출력 GP 모델을 상태 변수의 각 차원 별로 학습한다. 이를 위해, 식 (17)와 같이 시스템의 상태 변수 차원 n_x에 대해 n_x 개의 GP 모델을 구성하며, 각각의 학습 데이터 $D = \{Z, Y\}$를 차원별로 구분한다.

$$D_1 = \{Z_1 = [z_1, \cdots, z_j = [x_j, u_j], \cdots, z_n],$$
$$Y_1 = [\Delta x_2^1 \cdots, \Delta x_{j+1}^1, \cdots, \Delta x_{n+1}^1]\}$$

$$D_{n_x} = \{Z_{n_x} = [z_1, \cdots, z_j = [x_j, u_j], \cdots, z_n], \quad (17)$$
$$Y_{n_x} = [\Delta x_2^{n_x} \cdots, \Delta x_{j+1}^{n_x}, \cdots, \Delta x_{n+1}^{n_x}]\}$$

훈련된 모델은 MPC 제어기의 예측 루프에서 통합되어 모델의 예측과정에서 발생하는 오차를 추정하고, 불확실성을 반영한다. 이 과정을 통해 시스템 모델의 정확도를 높여 제어 성능을 개선한다.

2. 불확실성 전파

가우시안 프로세스와 MPC 예측 간에 통합에서 고려되어야 할 중요한 사항은 불확실성 전파(uncertainty propagation)이다. 모델 예측에는 현재 시점 k에서의 입력, 즉 측정값이 가우시안 분포를 따른 다고 가정한다. 하지만, 예측의 출력은 결정론적인 값이 아닌 확률적인 변수이며, 이는 명확한 분포를 가지고 있지 않다[33]. 즉, 가우시안 분포를 따르지 않을 수 있다. 이와 같은 불확실한 입력 z_k(현재 시간 단계의 예측 후 분포)을 다음 $k+1$ 단계의 상태를 예측하면 예측 오차가 누적된다. 예측한 상태 x_{k+1}의 평균과 분산은 다음과 같이 전개된다.

$$\mu_{k+1}^x = \mathbb{E}_{z_k}\left(f(x_k, u_k) + B_d \mu^d(x_k, u_k)\right) \quad (18)$$

$$\Sigma_{k+1}^x = \mathbb{V}_{z_k}\left(f(x_k, u_k) + B_d \mu^d(x_k, u_k)\right)$$
$$+ B_d \mathbb{E}_{z_k}\left(\Sigma^d(x_k, u_k)\right) B_d^T + B_d \Sigma^w B_d^T \quad (19)$$

식 (18), (19)와 같은 분포에 대한 계산은 연산 부하가 높고, 특히 여러 시간 단계에 걸친 시스템 동작 예측을 수행해야 하는 MPC 문제에서 연산량이 급격히 증가한다. 따라서 계산의 효율성을 높이기 위한 방법은 일반적으로 MPC의 예측 분포를 결정적으로 가정하여 입력의 분포를 가우시안 분포로 근사하는 것이다. 각 단계에서 예측 분포를 가우시안 분포로 근사하면 식 (20)과 (21)과 같이 평균과 공분산을 근사할 수 있다.

$$\mu_{k+1}^x \approx f(\mu_k^x, u_k) + B_d \mu^d(\mu_k^x, u_k) \quad (20)$$

$$\Sigma_{k+1}^x \approx \tilde{A}_k \Sigma_k^x \tilde{A}_k^T + B_d(\Sigma^d(\mu_k^x, u_k) + \Sigma^w) B_d^T \quad (21)$$

여기서 $\tilde{A}_k$ 는 정규 모델 f 의 Jacobian이며 현재 상태의 불확실성이 다음 단계로 전파되는 부분이다.

이러한 불확실성을 GP로 잔여 동역학을 학습한 모델 예측 제어는 다음과 같은 확률 예측 제어(stochastic predictive control) 문제로 표현할 수 있다[33,34].

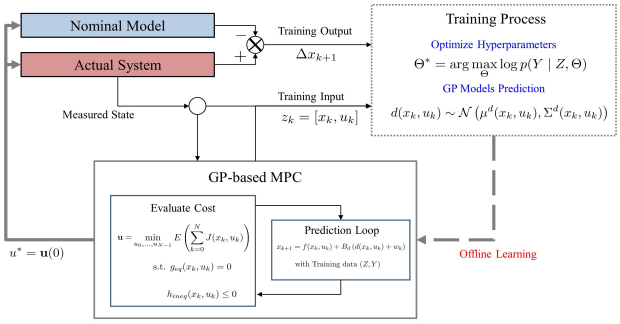

그림 3. GP–MPC를 통한 훈련 과정과 제어기 구조.

Fig. 3. Structure of control-loop and training process using GP–MPC.

$$\min_{\mathbf{u}} \mathbb{E}\left(\sum_{k=0}^{N-1} J(x_k, u_k)\right)$$
$$= \sum_{k=0}^{N-1} [(\mu_k^x - x_{\text{ref}})^T \mathbf{Q}(\mu_k^x - x_{\text{ref}}) + u_k^T \mathbf{R} u_k + \text{tr}(\mathbf{Q}\Sigma_k^x)] \quad (22)$$

$$\text{s.t.} \quad \mu_{k+1}^x = f(\mu_k^x, u_k) + B_d \mu^d(\mu_k^x, u_k), \quad (23a)$$

$$\Sigma_{k+1}^x = \tilde{A}_k \Sigma_k^x \tilde{A}_k^T + B_d(\Sigma^d(\mu_k^x, u_k) + \Sigma^w) B_d^T, \quad (23b)$$

$$p(x_{k+i|k} \geq x_{min}) \geq \beta_i, \quad (23c)$$

$$p(x_{k+i|k} \leq x_{max}) \geq \beta_i, \quad (23d)$$

$$u_{min} \leq u_{k+i-1} \leq u_{max}, \quad (23e)$$
$$i = 1, \ldots, N$$

식 (22), (23) 의 N은 예측 구간의 길이이며, $\mathbf{Q}, \mathbf{R}$은 상태와 입력에 대한 가중치 행렬이다.

가우시안 분포 가정 하에 확률 제약 조건을 결정론적으로 변환하는 일반적인 방법은 [35,36]에 제시되어 있다. 신뢰도 $\beta_i = 0.95$ 로 설정하면 상태 제약조건 (23c), (23d)는 다음과 식 (24)와 같이 결정적으로 나타낼 수 있다. 이러한 근사 방식은 확률 제약 조건에 비해 보수적인 조건을 보이게 된다.

$$\mu_{k+i}^x - 2 \cdot \Sigma_{k+i}^x \geq x_{min} \quad (24a)$$

$$\mu_{k+i}^x + 2 \cdot \Sigma_{k+i}^x \leq x_{max} \quad (24b)$$

이를 통해, GP–MPC는 근사화된 상태의 평균과 공분산을 사용하여 모델의 오차와 불확실성을 고려한 최적화 문제를 해결할 수 있다. GP 모델을 통합한 MPC 문제를 확률적 최적화 문제로 수식화함으로써, 모델 불확실성과 예측 오차를 고려할 수 있다. GP–MPC 프레임워크는 시스템 동역학에 대한 사전지식이 부족하거나 부분적으로 알려진 시스템에서 오차 데이터를 기반으로 제어 시스템의 성능을 향상한다.

IV. 시뮬레이션 검증

본 장에서는 가우시안 프로세스와 모델 예측 제어의 통합을 통해 불확실성이 있는 동적 시스템에서 제안한 알고리즘의 제어 성능을 검증한다. 실험 환경은 Intel i5-9400F 2.90 GHz, RAM 16GB로 MATLAB의 fmincon 솔버와 GPML Toolbox를 통해 제어기 점증을 진행하였다[37]. 본 시뮬레이션은 [38]

에서 제공된 자료를 변형하여 사용하였으며, 임의의 가우시안 노이즈를 추가하였다.[1] 시스템 모델에 대해 물리 기반 정규 모델을 활용한 nominal MPC의 결과와, 이를 통해 발생한 residual dynamics를 가우시안 프로세스로 학습하여 GP–MPC를 적용한 결과를 비교하였다.

1. 선형 시스템 제어 성능 검증

식 (25)은 차량의 평면 운동을 슬립각과 요레이트에 대한 상태 공간 방정식으로 나타낸 것으로 상태 변수 x는 횡방향 속도 및 요레이트, 입력 u을 조향각 δ를 사용하는 bicycle model이다.

$$\dot{x} = \begin{bmatrix} -\frac{C_f+C_r}{mv_x} & \frac{C_r l_r - C_f l_f}{mv_x^2} - 1 \\ \frac{C_r l_r - C_f l_f}{J_z} & -\frac{C_f l_f^2 + C_r l_r^2}{J_z v_x} \end{bmatrix} x + \begin{bmatrix} \frac{C_f}{mv_x} \\ \frac{C_f l_f}{J_z} \end{bmatrix} u \qquad (25)$$

차량의 속도 v_x가 일정하다고 가정하면, 식 (25)은 선형 시불변 모델이 되며, m, J_z, l_f, l_r은 각각 차량의 질량, 관성 모멘트, 전륜 및 후륜 축거를 나타낸다. C_f, C_r는 차량의 코너링 강성 계수로, 본 논문에서는 GP–MPC의 성능을 확인하기 위해 실제 시스템 모델과 MPC의 예측 모델 간에 코너링 강성 계수를 약 30%의 오차를 부여했다. 제어 목표는 요레이트 레퍼런스를 추종하면서 슬립각을 최소화하는 것을 목표로 한다.

MPC 모델의 예측 단계 동안 공분산의 불확실성 전파를 모두 고려하는 것은 계산 부하가 크므로 이전 장에서 논의한 바와 같이 식 (21)과 같은 방법으로 공분산을 계산하였다. 또한, 2장에서 언급한 sparse 기법 중 FITC 방법을 적용하여 SGP–MPC (Sparse GP–MPC)와의 결과도 함께 비교하였다.

그림 4는 17-21초 시간 동안 nominal MPC를 통해 수집한 Δx_{k+1} 데이터와 훈련된 GP 모델 d의 예측 결과를 비교한 그림이다. 대부분의 훈련 데이터가 예측 결과의 신뢰 구간에 포함됨을 알 수 있다. 그림 5는 각 제어기의 결과를 나타낸 그래프로 nominal MPC에 대비 GP–MPC는 요레이트 추종

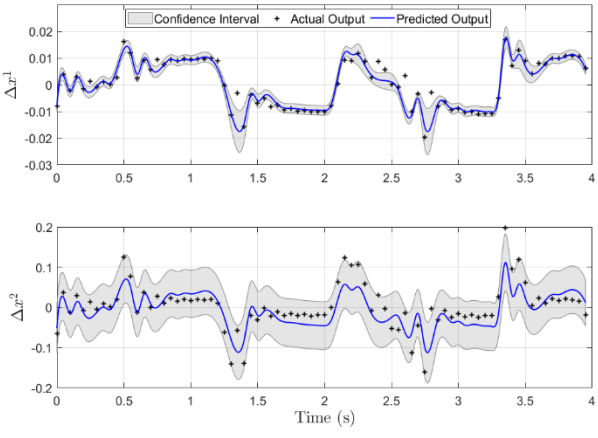

그림 4. Bicycle model에서 수집된 모델 오차와 GP 모델 예측 (95% 신뢰 구간 포함).

Fig. 4. Nominal model error and GP model prediction with 95% confidence interval in a bicycle model.

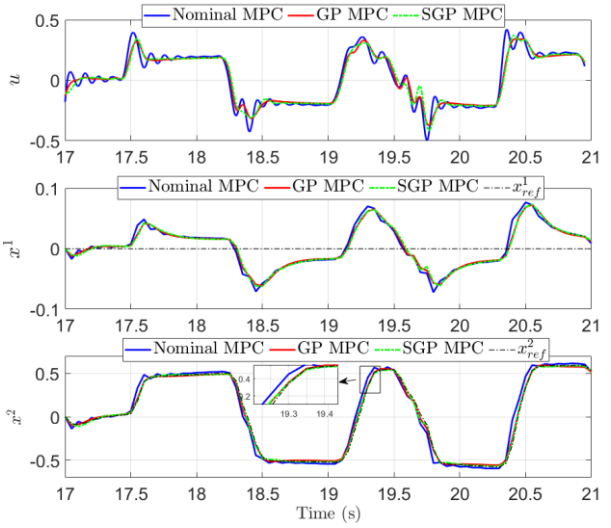

그림 5. Bicycle model에서 nominal MPC와 GP–MPC, SGP–MPC 제어 결과 비교.

Fig. 5. Control results comparison between nominal MPC, GP–MPC and SGP–MPC in a bicycle model.

표 1. 선형 시스템 연산 시간 비교.

Table 1. Computation time comparison in linear system.

	mean	max	std
Nominal MPC	9.3ms	23ms	4.1ms
GP–MPC	661ms	880ms	81ms
SGP–MPC	213ms	318ms	61ms

에러가 62.32% 감소 및 슬립각 에러는 5.26% 감소하였다. 이는 그림4와 같이 데이터 수집을 통해 GP 모델을 수립 및 이를 MPC 제어에 활용한 성능 개선이다.

SGP–MPC는 nominal MPC에 대비 요레이트 추종 에러는 58% 감소 및 슬립각 에러는 4.46% 감소하였다. 표 1은 제어기 간의 연산 시간을 나타낸 것으로 시뮬레이션 루프 동안 평균 연산 시간, 최대 연산 시간, 표준 편차를 계산하였다. Nominal MPC의 경우 평균적으로 10ms 이내의 연산

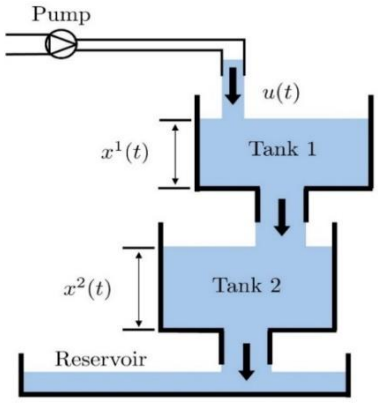

그림 6. 2단 물탱크 시스템.

Fig. 6. Two-tank system.

[1]본 연구에서 사용된 코드는 요청에 따라 제공가능하며, 기관 및 윤리 지침을 따릅니다.

속도를 보였다. 이에 반해 GP–MPC는 평균적으로 600ms가 넘는 연산속도를 보여주었으며 SGP–MPC는 GP–MPC 대비 약 67% 감소한 200ms 대의 속도를 보여주었다. 이는 sparse 기법을 통해 추종 정확도가 미미하게 저하되나 연산 시간에서 큰 개선을 보여주었다.

2. 비선형 시스템 제어 성능 검증

그림 6과 같이 2개의 다단 물탱크 시스템에 대하여 베르누이 방정식(Bernoulli's equation)을 활용하여 출구의 유속과 단면적을 기반으로 각 탱크 수위에 따른 유출량을 계산하면, 식 (26)과 같은 상태 방정식을 유도할 수 있다.

$$\frac{dx^1}{dt} = \frac{1}{A_1}\left(u - \sqrt{2gx^1 A_{\text{out1}}}\right) \tag{26a}$$

$$\frac{dx^2}{dt} = \frac{1}{A_2}\left(\sqrt{2gx^1 A_{\text{out1}}} - \sqrt{2gx^2 A_{\text{out2}}}\right) \tag{26b}$$

x^1 과 x^2 는 각각 탱크에 채워진 물의 높이이며, 제어 입력값 u 는 tank 1으로 유입되는 유량이다. 수위에 따른 유출량이 비선형적인 관계를 이루며, 이때, 입력은 최대 유입량 제한이 있으며, 각 탱크의 물의 높이는 탱크의 높이의 제약이 있다. A_1, A_2는 각 탱크의 단면적이며 A_{out1}, A_{out2}는 각 탱크의 유출 단면적이다. 해당 모델은 식 (27)의 4차 Runge-Kutta 방법을 사용하여 이산화하였다.

$$x_{k+1} = x_k + \frac{T_s}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{27}$$

이때 식 (26)을 실제 시스템으로 하고 MPC의 예측 모델에는 실제 시스템 모델과 약간의 파라미터 변화를 포함하여 사용한다. 탱크의 실제 유량의 유출량은 유체의 특성 및 유동 상태, 측정 오류 등으로 인해 불확실성과 변동성이 있다. 본 상황에서는 여러 요인으로 발생하는 불확실성을 간단화하기 위하여 유량의 유출 면적에 10%의 오차를 주어 실제 모델과 차이를 두었다. 그림 7은 nominal MPC를 통해 수집한 Δx_{k+1} 데이터와 훈련된 GP 모델 d 의 예측 결과를 비교한 그림이다. 이 결과는 nominal model과 실제 모델 사이의 residual dynamics를 GP 모델이 적절히 식별할 수 있음을 나타낸다.

그림 8은 다단 물탱크 시스템에 대한 각 제어기의 결과를 나타낸 그래프이다. Nominal MPC의 경우 약 16초 이후 거의 일정한 제어 입력을 가지지만, x^1에 대하여 일부 steady-state 오차가 발생하는 것을 확인할 수 있다. x^2 에서도 약 80초까지 레퍼런스를 추종하고 수위를 유지하는 이후에서도 일부 오차가 발생하는 것을 보여준다. GP–MPC의 경우, nominal MPC에 비해 상태 변수 x^1 에서 46.5% 더 낮은 RMSE 에러를 보여주며, x^2 에서 상승 구간에서 nominal MPC보다 느린 반응성을 보여주지만 레퍼런스 추종 정확도는 향상된 결과를 보여준다. SGP–MPC는 nominal MPC에 대비 x^1에 대한 에러가 44.50% 감소하였다. 표 2에서 제어기간의 연산 시간의 경우, nominal MPC는 평균적으로 8ms의 연산 속도를 보였다. 이전 선형 시스템 시뮬레이션에 비해 연산 속도가 빠른 것은 예측 구간의 값이 더 짧기 때문이다. 이에 반해 GP–MPC는 평균적으로 217ms, SGP–MPC는 GP–MPC 대비 약 64% 감소한 77ms 대의 속도를 보여주었다.

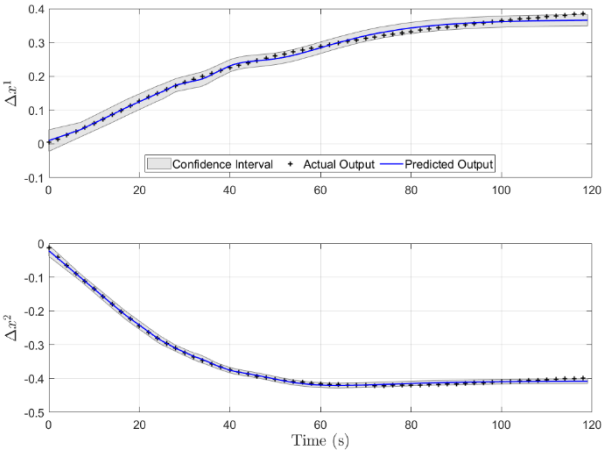

그림 7. Two-tank system에서 수집된 모델 오차와 GP 모델 예측 (95% 신뢰 구간 포함).

Fig. 7. Nominal model error and GP model prediction with 95% confidence interval in a two-tank system.

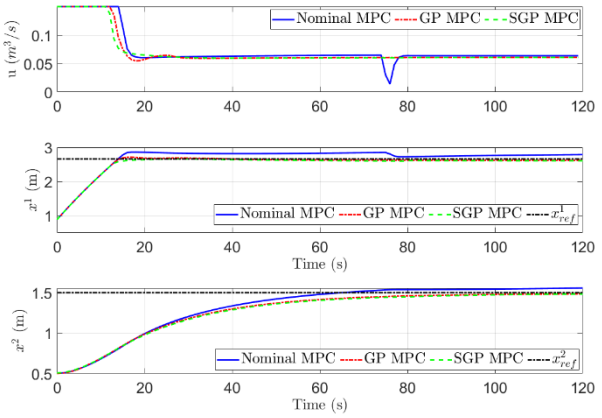

그림 8. Two-tank system에서 nominal MPC와 GP–MPC, SGP–MPC 제어 결과 비교.

Fig. 8. Control results comparison between nominal MPC, GP–MPC and SGP–MPC in a two-tank system.

표 2. 비선형 시스템 연산 계산 시간 비교.

Table 2. Computation time comparison in nonlinear system.

	mean	max	std
Nominal MPC	8ms	11.6ms	0.9ms
GP–MPC	217ms	473ms	77.1ms
SGP–MPC	77ms	106ms	11.5ms

V. 결론

본 논문에서는 가우시안 프로세스와 모델 예측 제어의 통합을 통해 복잡한 동적 시스템에서의 제어 성능을 향상시키는 방법에 대해 논의하였다. GP는 비선형 시스템에서 명시적인 모델 없이도 학습을 가능하게 하며, 이를 통해 MPC는 개선된 모델을 바탕으로 최적 제어 정책을 생성한다. 특히 GP의 높은 계산 부하를 감소하기 위해 sparse GP와 같은 근사 기법을 도입하여 계산 효율성에 대해서 논의하였다. GP 기반 MPC가 비선형 시스템 제어에서 기존의 nominal model 기반 방법보다 더 나은 성능을 발휘할 수

있음을 실험적으로 입증하였다. 또한, 다단계 예측에서 평균과 분산을 근사하는 방법을 통해 동적 환경에서의 불확실성 처리를 개선하였다. 다만, 본 연구는 시뮬레이션 기반 데이터로 GP 모델을 학습하였기 때문에, 실제 시스템에 적용하는 과정에서 발생할 수 있는 sim-to-real 이슈를 해결해야 할 필요가 있다. 시뮬레이션 환경에서는 노이즈나 데이터 오염 정도를 임의로 설정하기 쉽지만, 실제 하드웨어에서는 센서 측정 오차나 외란 형태가 달라질 수 있으므로 학습 데이터의 품질이 제어 성능에 미치는 영향을 면밀히 검토할 필요가 있다. 따라서, 향후 연구에서는 도메인 적응 기법, 실제 시스템을 반영한 데이터 보강, 그리고 GP 모델의 온라인 학습을 통해 실시간 환경에서도 제어 성능을 유지할 수 있는 방안을 제시할 필요가 있다.

본 논문에서 소개한 GP–MPC 접근 방식은 이론적 이해와 함께 학습 기반 제어 시스템의 실용성을 강조하며, 불확실성이 존재하는 시스템에서 효과적으로 적용될 수 있음을 보여주었다. 이를 통해 시뮬레이션을 넘어 실제 시스템에 적용 가능성을 높이는 연구 방향을 제시하며, 학습 기반 제어 분야의 연구 방향과 응용 가능성을 제공한다.

REFERENCES

[1] M. Schwenzer, M. Ay, T. Bergs, and D. Abel, “Review on model predictive control: An engineering perspective,” *The International Journal of Advanced Manufacturing Technology*, vol. 117, no. 5, pp. 1327-1349, 2021.

[2] Y. I. Lee, “Introduction of model predictive control strategies for control of power converters,” *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 30, no. 4, pp. 473-478, 2024. doi: 10.5302/J.ICROS.2024.24.0050.

[3] W. Langson, I. Chryssochoos, S. Raković, and D. Q. Mayne, “Robust model predictive control using tubes,” *Automatica*, vol. 40, no. 1, pp. 125-133, 2004.

[4] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, “Constrained model predictive control: Stability and optimality,” *Automatica*, vol. 36, no. 6, pp. 789-814, 2000.

[5] M. Farina, L. Giulioni, and R. Scattolini, “Stochastic linear model predictive control with chance constraints–a review,” *Journal of Process Control*, vol. 44, pp. 53-67, 2016.

[6] J. Mattingley and S. Boyd, “CVXGEN: a code generator for embedded convex optimization,” *Optimization and Engineering*, vol. 13, no. 1, pp. 1–27, 2012.

[7] B. Stellato, G. Banjac, P. Goulart, A. Bemporad, and S. Boyd, “OSQP: An operator splitting solver for quadratic programs,” *Mathematical Programming Computation*, vol. 12, no. 4, pp. 637-672, 2020.

[8] A. Bemporad and M. Morari, “Control of systems integrating logic, dynamics, and constraints,” *Automatica*, vol. 35, no. 3, pp. 407-427, 1999.

[9] B. K. Irdmousa, S. Z. Rizvi, J. M. Veini, J. Nabert, and M. Shahbakhti, “Data-driven modeling and predictive control of combustion phasing for RCCI engines,” *American Control Conference (ACC)*, pp. 1617-1622, 2019.

[10] D. G. Nguyen, S. Park, J. Park, D. Kim, J. S. Eo, and K. Han, “An mpc approximation approach for adaptive cruise control with reduced computational complexity and low memory footprint,” *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 2, pp. 3154–3167, 2024.

[11] X. Zhang, M. Bujarbaruah, and F. Borrelli, “Safe and near-optimal policy learning for model predictive control using primal-dual neural networks,” *American Control Conference (ACC)*, pp. 354-359, 2019.

[12] K. P. Wabersich and M. N. Zeilinger, “Safe exploration of nonlinear dynamical systems: A predictive safety filter for reinforcement learning,” arXiv:1812.05506, 2018.

[13] M. Klaučo, M. Kalúz, and M. Kvasnica, “Machine learning-based warm starting of active set methods in embedded model predictive control,” *Engineering Applications of Artificial Intelligence*, vol. 77, pp. 1-8, 2019.

[14] R. Kohút, L. Galčíová, K. Fedorová, T. Ábelová, M. Bakošová, and M. Kvasnica, “Hidden Markov model-based warm-start of active set method in model predictive control,” *2021 23rd International Conference on Process Control (PC)*, pp. 60-65, 2021.

[15] D. Masti and A. Bemporad, “Learning nonlinear state–space models using autoencoders,” *Automatica*, vol. 129, p. 109666, 2021.

[16] K. H. Kim, C. Jeong, J. Kim, S. Lee, and C. M. Kang, “Neural network modeling approach for vehicle lateral dynamics using real vehicle data,” *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 30, no. 7, pp. 696-702, 2024. doi: 10.5302/J.ICROS.2024.24.0082.

[17] L. A. Basina, B. K. Irdmousa, J. M. Velni, H. Borhan, J. D. Naber, and M. Shahbakhti, “Data-driven modeling and predictive control of maximum pressure rise rate in RCCI engines,” *2020 IEEE Conference on Control Technology and Applications (CCTA)*, pp. 94-99, 2020.

[18] L. Hewing, K. P. Wabersich, M. Menner, and M. N. Zeilinger, “Learning-based model predictive control: Toward safe learning in control,” *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 3, no. 1, pp. 269-296, 2020.

[19] A. Kamath, R. A. Vargas-Hernández, R. V. Krems, T. Carrington, and S. Manzhos, “Neural networks vs Gaussian process regression for representing potential energy surfaces: A comparative study of fit quality and vibrational spectrum accuracy,” *The Journal of Chemical Physics*, vol. 148, no. 24, 2018.

[20] J. Kocijan, *Modelling and Control of Dynamic Systems Using Gaussian Process Models*. Springer, 2016.

[21] P. Xu, H. Qin, J. Ma, Z. Deng, and Y. Xue, “Data-driven model predictive control for ships with Gaussian process,” *Ocean Engineering*, vol. 268, p. 113420, 2023.

[22] E. Snelson and Z. Ghahramani, “Sparse Gaussian processes using pseudo-inputs,” *Advances in Neural Information Processing Systems,* vol. 18, 2005.

[23] M. Titsias, “Variational learning of inducing variables in sparse Gaussian processes,” *Artificial Intelligence and Statistics*. PMLR, pp. 567-574, 2009.

[24] E. Bradford, L. Imsland, D. Zhang, and E. A. del Rio Chanona, “Stochastic data-driven model predictive control using gaussian processes,” *Computers & Chemical Engineering,* vol. 139, p. 106844, 2020.

[25] L. Hewing, J. Kabzan, and M. N. Zeilinger, “Cautious model predictive control using gaussian process regression,” *IEEE Transactions on Control Systems Technology,* vol. 28, no. 6, pp.

2736-2743, 2019.
[26] D. He and B. Peng, "Gaussian learning-based fuzzy predictive cruise control for improving safety and economy of connected vehicles," *IET Intelligent Transport Systems*, vol. 14, no. 5, pp. 346-355, 2020.
[27] M. Maiworm, D. Limon, and R. Findeisen, "Online learning-based model predictive control with Gaussian process models and stability guarantees," *International Journal of Robust and Nonlinear Control*, vol. 31, no. 18, pp. 8785-8812, 2021.
[28] C. E. Rasmussen, *Gaussian Processes in Machine Learning*, vol. 1. Cambridge, MA, USA: MIT Press, 2006.
[29] J. Jihoon, Y. Donggil, and O. Junghyun, "Three-dimensional radiation mapping using gaussian process regression and octomap," *Journal of Institute of Control, Robotics and Systems (in Korean),* vol. 29, no. 4, pp. 320-324, 2023. doi: 10.5302/J.ICROS.2023.22.8010.
[30] E. Schulz, M. Speekenbrink, and A. Krause, "A tutorial on Gaussian process regression: Modelling, exploring, and exploiting functions," *Journal of mathematical psychology,* vol. 85, pp. 1-16, 2018.
[31] M. Bauer, M. Van der Wilk, and C. E. Rasmussen, "Understanding probabilistic sparse Gaussian process approximations," *Advances in Neural Information Processing Systems,* vol. 29, 2016.
[32] H. Bijl, J.-W. van Wingerden, T. B. Schön, and M. Verhaegen, "Online sparse Gaussian process regression using FITC and PITC approximations," *IFAC-PapersOnLine,* vol. 48, no. 28, pp. 703-708, 2015.
[33] J. Wang, M. T. Fader, and J. A. Marshall, "Learning-based model predictive control for improved mobile robot path following using Gaussian processes and feedback linearization," *Journal of Field Robotics,* vol. 40, no. 5, pp. 1014-1033, 2023.
[34] A. Mesbah, "Stochastic model predictive control: An overview and perspectives for future research," *IEEE Control Systems Magazine,* vol. 36, no. 6, pp. 30-44, 2016.
[35] M. Cannon, B. Kouvaritakis, and J. Primbs, "Stochastic model predictive control," *Proc. of 17th International Federation of Automatic Control (IFAC) World Congress*, Seoul, Korea, pp. 6-11, Jul. 2008.
[36] S. V. Raković, "Model predictive control: classical, robust, and stochastic [bookshelf]," *IEEE Control Systems Magazine,* vol. 36, no. 6, pp. 102-105, 2016.
[37] C. E. Rasmussen and H. Nickisch, "The GPML toolbox version 4.0," Technical Documentation, 2016.
[38] M. Pfefferkorn and R. Findeisen, "Gaussian processes in model predictive control: Basics, safety, and stability," *Proc. of 22nd International Federation of Automatic Control (IFAC) World Congress*, Yokohama, Japan, 2023.

김 정 효

2023년 경북대학교 기계공학부 졸업. 2023년~현재 한양대학교 대학원 미래자동차공학과 석사과정 재학 중. 관심분야는 차량 제어, 상태 추정, 최적제어.

박 수 용

2021년 영남대학교 기계공학부 졸업. 2024년 경북대학교 대학원 기계공학과 석사. 2024~현재 한양대학교 대학원 미래자동차공학과 박사과정 재학 중. 에너지 최적화, 커넥티드 및 자율 주행.

한 경 석

2013년 한양대학교 건설환경공학과 졸업. 2015년 한국과학기술원 기계공학전공 석사. 2018년 동 대학원 공학박사. 2020년~2024년 경북대학교 기계공학부 부교수. 2024~현재 한양대학교 미래자동차공학과 부교수. 관심분야는 차량 제어, 에너지 최적화, 강화 학습, 최적 제어.