

비정돈 환경의 표면 소독을 위한 실현성 예측 기반의 장애물 제거 계획법 및 접촉식 방역 로봇 시스템

Feasibility Prediction-Based Obstacle Removal Planning and Contactable Disinfection Robot System for Surface Disinfection in an Untidy Environment

강준수¹·이인제²·정완균³·김기훈[†]

Junsu Kang¹, Inje Yi², Wan Kyun Chung³, Keehoon Kim[†]

Abstract: We propose a task and motion planning algorithm for clearing obstacles and wiping surfaces, which is essential for surface disinfection during the pathogen disinfection process. The proposed task and motion planning algorithm determines task parameters such as grasping pose and placement location during the planning process without using pre-specified or discretized values. Furthermore, to quickly inspect many unit motions, we propose a motion feasibility prediction algorithm consisting of collision checking and an SVM model for inverse mechanics and self-collision prediction. Planning time analysis shows that the feasibility prediction algorithm can significantly increase the planning speed and success rates in situations with multiple obstacles. Finally, we implemented a hierarchical control scheme to enable wiping motion while following a planner-generated joint trajectory. We verified our planning and control framework by conducted an obstacle-clearing and surface wiping experiment in a simulated disinfection environment.

Keywords: Task and Motion Planning, Manipulation Planning, Hierarchical Control, Surface Disinfection

1. 서 론

유래없이 강력한 전염성과 고령층 치사율을 동시에 보인 COVID-19 팬데믹 위기가 장기화됨에 따라, 일상 환경에서의 방역 서비스에 대한 수요가 폭증하고 방역 관련 인프라 구축에 대한 관심이 높아지고 있다^[1]. 이에 따라 여러 기업과 연구기관에서 방역 로봇을 개발해 선보이고 있는데, 국제 및 국내 방역 지침은 에어로졸 흡입 위험 및 불확실한 소독 효과를 이유로 분사식 소독과 UV 조사를 지양하고 직접적으로 표면을 닦는 방법을 권고하는 반면^[2,3] 현재 개발된 방역 로봇들은 소

독액의 분사 및 UV 조사를 통한 간접적 방식에 머물고 있어 국제 및 국내 방역 지침을 충족시키지 못하고 있다^[1]. 이전에 개발된 로봇들이 표면을 직접 닦는 접촉식 방역 작업을 수행하지 못하는 원인은, 우선 표면 소독을 위해서는 표면 상에 놓인 장애물을 움직여 사각지대를 제거할 수 있어야 하는데 일상 환경의 비정형화 된 장애물에 대응하는 데 한계가 있다는 점, 그리고 접촉 상태를 효과적으로 유지하기 위해 상용 로봇이 흔히 탑재하고 있지 않은 힘제어가 필요한 점 등이 있다.

상기한 두 이유 중 장애물에 의한 사각지대 제거 문제는 닦기 동작과 물체 이동을 포함한 일종의 Task and Motion Planning (TAMP) 문제로 볼 수 있다. TAMP는 특정 순서로 여러 물체를 이동시키는 것과 같은 작업 수준의 계획과 해당 동작을 수행하기 위한 구체적 경로의 생성이 동시에 이루어져야 하는 문제로, 작업 수준과 동작 수준의 계획이 완전한 계층 구조로 나눠지지 않고 서로 영향을 미치기 때문에 단방향으로 해답을 구할 수 없어서 많은 연산량을 요구하는 문제이다. 기존의 많은 연구들이 TAMP의 연산 시간을 단축하기 위해 노력해왔는데,

Received : May. 20. 2021; Revised : Jun. 7. 2021; Accepted : Jun. 7. 2021

※ This research was supported by Korea Advanced Research Program through the National Research Foundation of Korea (NRF) funded by Ministry of Science and ICT (2020M3H8A1114905)

1. Ph.D. Student, POSTECH, Pohang, Korea (junsu_kang@postech.ac.kr)

2. Master Student, POSTECH, Pohang, Korea (inje@postech.ac.kr)

3. Professor, POSTECH, Pohang, Korea (wkchung@postech.ac.kr)

† Associate Professor, Corresponding author: Mechanical Engineering, POSTECH, Pohang, Korea (khk@postech.ac.kr)

크게 작업 수준의 계획을 효과적으로 업데이트 하기 위한 방법론과¹⁴ 학습을 통해 작업 수준 계획을 예측하는 방식^{7,8}, 그리고 동작의 실현성 예측을 통해 불가능한 동작을 빠르게 배제하는 방식^{9,10} 있다. 앞선 두 방식 역시 연산 과정 효율화에 필수적 요소이나, TAMP에 소요되는 시간의 대부분이 동작 계획, 특히 실현 불가능한 동작의 계획 시도에 소요된다는 점은 이전의 많은 연구에서 지적되어 온 사실이다^{6,9}. 이에 따라, 본 연구에서는 불가능한 동작의 배제에 초점을 맞춰 장애물 치우기 및 닦기로 구성된 방역 작업에서 불가능한 동작을 빠르게 배제할 수 있는 작업 계획 생성 알고리즘을 제안하고자 한다. 특히, 이전의 학습 기반 예측 방식들^{9,10} 과 달리, 본 논문의 방법은 작업을 이산화하거나 특정 환경 구조를 기반으로 하지 않아 새로운 환경 및 대상물에도 추가적인 학습 없이 적용이 가능하다. 또한, 제안된 시스템의 실현성을 검증하기 위해 생성된 방역 작업 계획에 접촉력 제어를 적용하고 실제 듀얼 암 테스트베드에서 시험하여 능동적 장애물 제거를 포함한 접촉식 방역 로봇 시스템의 실현 가능성을 보이고자 한다.

2. 접촉식 방역 작업

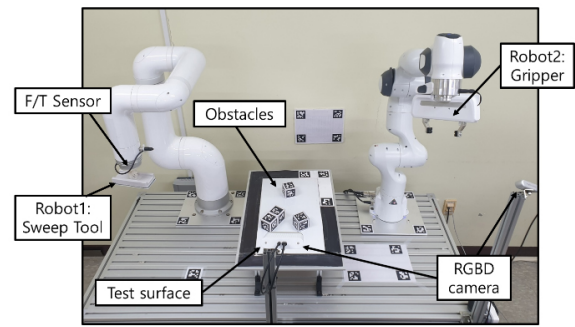
본 장에서는 본 논문이 목표로 하는 접촉식 방역 작업과 모의 실험 환경에 대해 서술한다.

2.1 접촉식 방역 작업

국내 및 국제 방역 지침은 소독제로 천을 적신 후 손길이 닿는 표면을 직접 닦아내는 방식의 접촉식 방역을 권고하고 있다^{2,3}. 이를 위해서는 닦기 작업을 수행하기 전 닦을 대상 경로의 장애물을 제거하는 작업이 선행될 필요가 있다. 이때, 실제 방역 환경에서는 장애물 수에 제한이 없고 로봇의 그리퍼는 한정되어 있기 때문에, 복수의 장애물을 제거하기 위해서는 장애물을 적절한 위치에 내려놓는 과정도 반드시 필요하다. 따라서 본 논문에서 다루는 접촉식 표면 방역 작업은 방역 대상면을 몇 개의 구획으로 나누고, 각 구획의 이동 경로상에 있는 장애물을 방해가 되지 않는 위치로 이동시킨 뒤 닦는 작업을 반복해 대상 표면 전체를 닦는 것으로 정의될 수 있을 것이다.

2.2 모의 방역 환경

본 논문에서는 표면 방역 작업을 단순화하여 [Fig. 1]과 같은 모의 방역 테스트 환경을 구성했다. 테스트베드에는 6 DoF 로봇 암(Indy7, Neuromeka)과 7 DoF 로봇 암(Panda, Franka Emika)이 한 대씩 설치되었다. 6 DoF 로봇 말단에는 닦기 작업을 위한 너비 약 14 cm의 Wiper와 F/T 센서가 설치되었고, 7



[Fig. 1] Testbed for surface wiping task with obstacles

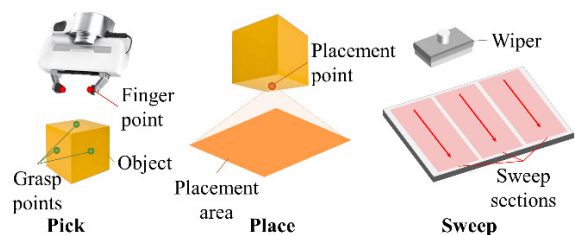
DoF 로봇의 말단에는 장애물 이동을 위한 2조 그리퍼가 설치되었다. 방역 대상면으로는 화이트보드를 배치하고, 중앙 30×50 cm² 범위를 목표 영역으로 설정했다. 모의 장애물로는 5×5×5 cm³, 10×5×5 cm³의 직육면체가 대상면 위 임의의 위치에 배치되었다. 이러한 모의 방역 환경에서, 본 논문의 목표는 임의로 배치된 장애물들을 적절한 위치로 옮겨가며 대상 방역면 전체를 닦는 작업을 수행하는 것이다. 단, 여러 형태의 비정형화된 방역면과 장애물이 존재하는 실제 일상 환경에도 별도의 사전 설정 없이 쉽게 적용될 수 있도록, 작업 변수(물체를 잡는 자세와 놓는 위치 등)의 사전 설정 및 이산화를 배제하고 작업 계획 과정에서 모든 변수가 스스로 결정될 수 있도록 한다.

3. 방역 작업 계획

2.1에서 정의된 접촉식 표면 방역 작업은 로봇의 자유 동작, 물체를 파지한 상태로 이동하는 동작, 그리고 방역 도구면과 방역 대상면 사이에 접촉 구속조건이 추가되는 닦기 동작으로 구성된다. 본 장에서는 이러한 동작으로 구성된 방역 계획을 생성하는 작업 및 동작 계획 알고리즘에 대해 서술한다.

3.1 방역 작업 정의

본 논문에서는 방역 작업을 이루는 각 동작을 J. Mirabel and F. Lamiriaux의 연구와 유사하게 구속 조건을 통해 정의하고¹¹, 각 동작을 임의로 시도하는 트리 탐색을 통해 작업 계획을 생성한다¹². 표면 방역 작업을 구성하는 각 단위 동작은 [Fig. 2]에 그림으로 나타나 있으며, 아래에서는 각 단위 동작의 정의를 서술한다.



[Fig. 2] Components of the surface disinfection task

3.1.1 물체의 파지

각 물체는 너비가 7 cm 이하인 모든 면을 따라 파지 영역이 정의되었다. 이때 파지 영역은 깊이 2.5 cm, 앞 뒤 여유 값 2.5 cm를 갖도록 하고 파지 각도는 면에 수직인 방향을 기준으로 $\pm 45^\circ$ 범위에서 접근이 가능하도록 했다. 즉, $5 \times 5 \times 5 \text{ cm}^3$ 정육면체의 경우 중심점을 파지 점으로 하여 6 면에서 접근이 가능하고, 그리퍼의 방향은 4 방향이 가능하기 때문에 총 24개의 파지 방식이 정의되며, 각 파지 방식은 $\pm 45^\circ$ 의 각도 변수를 갖게 된다. $10 \times 5 \times 5 \text{ cm}^3$ 직육면체의 $10 \times 5 \text{ cm}^2$ 인 4 면의 경우, 10 cm 축을 따라 파지 영역이 정의되고, 앞 뒤 여유 값을 제외하고 5 cm 길이의 파지 영역이 정의되어 1차원 위치 변수가 추가된다. 물체를 파지하기 위한 접근 동작은 위와 같이 정의된 파지 영역에서 임의의 변수로 파지 위치 및 방향을 특정한 뒤, 해당 지점에 그리퍼의 끝점을 위치시키는 것으로 정의된다.

3.1.2 물체의 배치

인식 시점에 각 물체에서 아래쪽을 향하는 면이 바닥 면으로 정의되며, 이 바닥 면의 중심을 방역 대상 면에 일치시키는 것으로 물체의 배치(Place) 동작이 정의된다. 대상 면 전체가 물체를 놓을 수 있는 위치가 됨에 따라, 2차원 위치에 수직 방향 회전각까지 총 3개의 변수를 갖게 된다. 따라서 물체를 놓는 동작을 위해서는 이 3개의 변수에 대한 결정이 필요하게 된다.

3.1.3 닦기 동작

본 논문에서는 방역 대상 영역을 방역 도구의 너비로 나눈 직선 구획의 조합으로 닦기 작업을 정의하였다. 제시된 실험에서는 30 cm 너비의 방역 면을 약 14 cm의 도구로 닦아야 하기 때문에 최소 3회의 직선 동작이 필요하다. 이에 따라 대상 표면을 3 개의 구획으로 균등하게 나누었으며, 각 구획의 닦기 동작은 구획의 시작 점에 방역 도구를 일치시키는 접근 동작과 해당 구획의 선 구속조건을 따라 구획의 종료 점까지 방역 도구를 이동시키는 닦기 동작으로 구성된다.

3.1.4 홈 회귀 동작

실험에 사용된 작업 공간은 매우 좁기 때문에, 두 로봇이 동시에 작업하는 것이 불가능한 상황이 자주 발생하게 된다. 따라서 다른 로봇의 작업을 위해 작업 공간을 비워줄 수 있도록 홈 자세로의 회귀 동작이 정의되었다.

3.2 작업 및 동작 계획 프레임워크

위의 작업 정의에 따라, 장애물의 파지와 이동에는 각각 1개 혹은 2개, 그리고 3개의 변수에 대한 결정이 필요하다. 여기서, 실제 일상 환경에 존재하는 임의의 방역 대상 면에 배치된 임의의 장애물에 대해서는 어떤 변수가 유효한 변수인지 미리

Algorithm 1 Task and Motion Planning

```

Input: Initial state  $\sigma_0$ , target state family  $\Sigma_g$ , feasibility rule set  $\mathcal{R}$ , timeout  $T$ 
Output: Task and motion tree  $(\mathbb{V}, \mathbb{E})$ 
1:  $\mathbb{V} \leftarrow \{\sigma_0\}$  ▷ Tree nodes
2:  $\mathbb{E} \leftarrow \emptyset$  ▷ Transition edges
3: repeat
4:    $\sigma_{new}, \sigma_{pre} \leftarrow \text{SampleTransition}(\mathbb{V})$ 
5:   for  $r \in \mathcal{R}$  do ▷ Predict feasibility
6:      $feasible \leftarrow \text{CheckRule}(r, \sigma_{parent}, \sigma)$ 
7:     if not  $feasible$ 
8:       break for
9:   if  $feasible$  ▷ Try motion planning
10:     $Q \leftarrow \text{MotionPlan}(\sigma_{parent}, \sigma, \mathcal{R})$ 
11:    if  $Q$ 
12:       $V \leftarrow V \cup \{\sigma\}$ 
13:       $E \leftarrow E \cup \{(\sigma_{parent}, \sigma, Q)\}$ 
14: until  $\text{CheckGoal}(\sigma_{new}, \Sigma_g)$  or  $\text{TimeOut}(T)$ 
    
```

[Fig. 3] TAMP algorithm used in the paper. Given an initial state σ_0 , a random tree is propagated toward the target state family Σ_g . For each transition, the feasibility of each motion is tested based on feasibility rules \mathcal{R} before planning the exact motion

확인할 방법이 없고, 때로는 당장의 동작에는 유효한 변수이더라도 이후의 동작에는 유효하지 않은 경우도 있다. 예를 들어, 특정 자세로 장애물의 파지는 가능했다라도 적절한 위치에 놓는 것은 불가능할 수 있다. 이러한 장기적인 목표 달성을 위해 작업 및 동작 계획, 즉 TAMP 문제가 해결될 필요가 있다. 본 논문에서는 시작상태로부터 임의로 단위 동작의 계획 및 추가를 반복하며 트리 탐색을 진행해 작업 계획을 생성한다[Fig. 3]. 이는 K. Hauser and V. Ng-Thow-Hing의 Random-MMP과 유사한 방식이¹²⁾, 본 논문에서는 매 동작 계획 이전에 실현 불가능한 동작을 배제해 계획 속도를 높일 수 있도록 했다. 실현성 예측 이후 최종적인 동작 계획의 경우, 일반적인 경로 계획에는 RRT-Connect¹³⁾, 구속 동작의 계획에는 Projection 기반의 RRT 알고리즘이 적용되었다¹⁴⁾.

3.3 동작 실현성 예측

I. Rodriguez et al.의 연구에서 제시되었듯 동작 계획의 복잡도는 고려 범위에 따라 대상물 단계, 도구 단계, 로봇 단계로 나눌 수 있고 엄밀성에 따라 정적 수준, 운동학적 수준, 동역학적 수준으로 구분 될 수 있다¹⁵⁾. 본 연구에서는 테이블과 같이 3차원적 장애물이 많지 않은 정적 환경을 전제로 장애물 이동과 닦기 작업을 수행하려 하기 때문에 정적 수준에서 대상물, 도구, 로봇 단계의 실현성 예측만으로 충분하다고 판단했다. 아래에서는 각 단계의 실현성 예측 구현 방식을 서술한다.

3.3.1 대상물 및 도구 단계

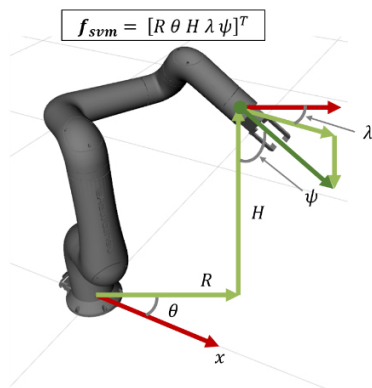
정적 수준에서 대상물, 도구 단계의 동작 실현성은 목표

위치에서 대상물, 도구, 주변 환경이 서로 충돌을 일으키지 않는지를 의미한다. 객체 간의 충돌 감지는 현재까지 다양한 알고리즘이 개발되어 있다. 본 논문에서는 Gilbert-Johnson-Keerthi (GJK)^[6] 알고리즘을 사용해 동작 목표 위치에서 객체 간의 충돌을 검사한다. 단, 닻기 동작과 장애물 놓기 동작의 경우는 현재 목표 구획 위 5 cm 영역과 장애물 사이의 충돌을 추가적으로 검사해 닻는 동작 도중에 발생할 충돌을 미리 방지하도록 했다. GJK 알고리즘은 기본적으로지만 매우 빠르고 효율적이라고 알려진 충돌 검사 알고리즘이기 때문에, 이 단계는 평균적으로 약 0.51 ms의 매우 짧은 시간 내에 완료될 수 있었다.

3.3.2 로봇 단계

정적 수준에서 로봇 단계의 동작 실현성은 자기 충돌 없이 목표 위치에 엔드툴이 위치하는 기구학적 해가 존재하는지를 의미한다. 이는 일반적으로 역기구학과 충돌 검사를 통해 확인할 수 있는데, 수치적 역기구학 해법은 수 ms 이상의 시간이 소요되는 반면 해석적 역기구학 해는 항상 가용하지 않기 때문에 문제가 된다. 또한, 여유 자유도가 있는 로봇의 경우 수많은 역기구학 해가 존재하기 때문에 복수의 역기구학 해에 대해 모두 자기 충돌을 검사해야 할 수 있어 많은 시간을 소요하게 될 우려가 있다. 때문에 본 논문에서는 자기 충돌을 고려한 역기구학 해의 존재 여부를 1 ms 이내의 빠른 시간 내에 확인하기 위해 support vector machine (SVM)을 이용했다.

구체적으로, 본 연구에서는 말단 조인트의 위치 및 방향 정보를 5 차원 특징 벡터로 변환하고 이를 기반으로 해의 존재 여부를 판단하는 SVM를 학습시켰다[Fig. 4]. 여기서 말단 조인트의 위치는 로봇의 첫번째 축과 정렬된 원통 좌표계에서 표현한 3차원 벡터로 표현했고, 방향은 말단 조인트의 회전축 벡터를 지평좌표계로 나타낸 2차원 벡터로 표현했다. 여기서 말단의 방향을 3차원 벡터로 나타내지 않은 것은 대다수 상용 로봇이 말단 조인트에 대해서는 360° 이상의 회전각을 보장하



[Fig. 4] 5D input feature vector for SVM model. R, θ, H for position of last joint and λ, ψ for direction of the last joint

[Table 1] Feasibility prediction accuracy for end-effector poses

Robot	GT	Accuracy (%)
6-DoF	Feasible	98.02
	Infeasible	98.59
	All	98.44
7-DoF	Feasible	95.29
	Infeasible	98.77
	All	98.16

기 때문에 마지막 조인트의 값은 정적인 자세의 실현 가능성에 영향을 미치지 않기 때문이다. SVM 학습에는 sci-kit-learn 라이브러리의 RBF kernel 구현을 사용했다^[7].

학습을 위한 데이터는 임의의 말단 자세를 위의 방식으로 기록하고 해당 자세로의 동작 계획을 시도해 결과를 기록하는 방식으로 수집했으며, 각 로봇에 대해 20,000 개의 데이터를 수집해 학습했다. 학습된 모델은 별도로 수집한 5,000 개의 테스트 데이터셋을 대상으로 테스트했으며 이에 대한 결과는 [Table 1]에 정리되어 있다. 두 로봇 모두에서 평균적으로 98% 이상의 정확도를 보였으며, SVM 예측의 평균 소요 시간은 0.16 ms였다.

위의 두 단계에 걸친 작업 실현성 예측은 작업을 이산화하여 정의할 필요가 없고 작업 환경에도 특정한 제약을 갖지 않는다. 즉, 임의의 파지 방향에 적용될 수 있고 작업 평면의 넓이, 높이 및 각도의 변화에도 영향을 받지 않아 임의의 환경에서 다양한 대상물 및 작업 환경에 적용이 가능하며, 환경에 따른 추가적인 학습이 불필요하다. 이는 기존의 방법들이^[9,10] 사전 정의된 2차원 평면에서 파지 방향에 따라 작업을 이산화하여 학습했던 점과 차별화되며, 실현성 예측 기반 TAMP의 활용성을 크게 개선했다고 할 수 있다.

4. 관절 동작 및 접촉력의 계층적 제어

방역 동작 중 닻기 동작의 경우 대상면에 수직 방향으로는 일정한 접촉력을 유지하면서 수평 방향의 특정 경로를 따라 이동하는 것으로 정의될 수 있으나, 실제 환경에서 방역 작업 시에는 주변 장애물에 대한 충돌 회피 역시 고려되어야 한다. 때문에 접촉력 방향과 분리가 가능한 작업 공간 상의 경로가 아닌, 장애물 회피를 고려한 관절 경로를 따라 작업이 정의될 필요가 있다. 하지만 작업 공간과 달리 관절 공간 상에서의 위치 제어는 접촉력 유지와 완전히 분리되지 않아, 두 가지 목표를 동시에 달성하기 위해서는 계층적 제어(Hierarchical control) 기법이 적용될 필요가 있다. 이하에서는 접촉력을 유지하면서 장애물을 피하는 관절 경로를 추종하기 위해 본 논문에서 적용된 계층적 제어 알고리즘을 서술한다.

4.1 계층적 제어

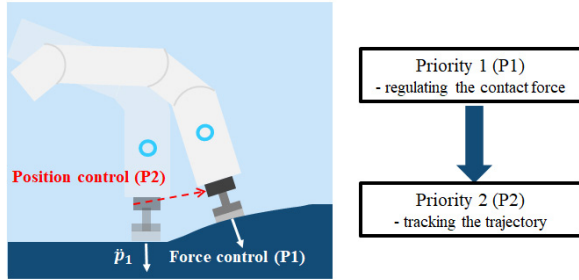
계층적 제어는 작업들 간의 우선순위를 선정한 후, 상위 계층의 작업들의 수행을 우선하여 동작시키는 제어 알고리즘을 의미한다. 로봇 시스템에서 계층적 제어는 기구학적 여유자유도를 활용하여 복합적인 작업들을 동시에 처리할 수 있게 한다. 본 연구에서는 [Fig. 5]와 같이 힘 제어를 상위, 위치 제어를 하위로 선정하여 두 작업을 동시에 수행했다.

4.2 관절 경로 추종

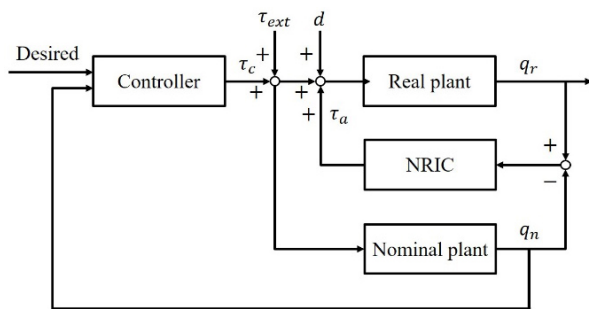
본 논문에서는 관절 경로 추종을 위해 [Fig. 6]와 같은 NRIC (Nonlinear Robust Internal-loop Compensator) 프레임워크 기반의 순응 제어를 적용한다^[18]. 이때, nominal model q_n 은 식 (1)과 같이 위치 에러로부터 관절공간 가속도 \ddot{q}_n 를 계산함으로써 업데이트 된다.

$$\ddot{q}_n = M_n^{-1}(-C_n \dot{q}_n + k_p e + k_v \dot{e} + \tau_{ext}) \quad (1)$$

k_p, k_v 는 Controller의 PD 게인, e 는 목표 값과 nominal model의 오차를 의미한다.



[Fig. 5] Proposed task hierarchy: priority 1 (force control) and priority 2 (position control in joint space)



[Fig. 6] The structure of NRIC framework with a controller

4.3 접촉력 제어

접촉력 제어 상황에서는 힘 제어와 관절 경로 추종을 동시에 수행하기 위해 앞서 정의한 동작 계층을 기반으로 nominal model의 가속도를 구한다.

계층으로 구분된 작업 공간을 확장된 자코비안 행렬로 표현하면 다음과 같다.

$$\begin{pmatrix} \dot{p}_1 \\ \dot{p}_2 \end{pmatrix} = \bar{J} \dot{q} = \begin{pmatrix} \bar{J}_1 \\ \bar{J}_2 \end{pmatrix} \dot{q} \quad (2)$$

\bar{J}_1 은 자코비안 행렬에서 접촉 방향 성분만을 포함한 행렬, \bar{J}_2 는 \bar{J}_1 의 영공간으로 span 되어진 자코비안 행렬, \dot{p}_2 는 상위 작업의 영공간에 투영된 하위 작업의 속도를 의미한다^[19].

식 (2)를 미분하면 관절 공간에서의 가속도를 구할 수 있다.

$$\ddot{q} = -\bar{J}^{-1} \dot{\bar{J}} \dot{q} + \bar{J}^{-1} \ddot{p} \quad (3)$$

뉘기 작업 시 로봇의 속도가 빠르지 않으므로 본 논문에서는 $\dot{\bar{J}}$ 항을 무시하고 아래와 같이 근사하여 적용하기로 한다.

$$\ddot{q} = \bar{J}^{-1} \ddot{p} \quad (4)$$

접촉력 제어를 위해, 본 논문에서는 F/T 센서를 통해 측정된 접촉 방향의 normal force 오차 e_f 를 nominal model의 해당 방향 가속도 \ddot{p}_1 에 PID 형식으로 피드백하는 방식을 택했다.

$$\ddot{p}_1 = K_p e_f + K_d \dot{e}_f + K_I \int e_f \quad (5)$$

접촉 상태에서 nominal model이 실제 위치에서 벗어남에 따라, NRIC 프레임워크에 의해 해당 방향으로 힘이 발생하게 된다. \ddot{p}_1 과 위치 제어에서 구한 가속도 \ddot{q}_n 을 식 (4)에 적용하면 접촉력 제어와 경로 추종이 모두 고려된 nominal model 가속도를 다음과 같이 얻을 수 있다^[19].

$$\ddot{q}_n' = \bar{J} \ddot{p} = J_1^M \ddot{p}_1 + Z_2^T \bar{J}_2 \ddot{q}_n \quad (6)$$

J_1^M 는 \bar{J}_1 의 가중 유사 역행렬, Z_2 는 \bar{J}_1 의 영공간 기저 행렬에 해당한다.

제어기 적용시 nominal model의 가속도에 위치 제어가 구현된 식 (1)과 힘 제어가 포함된 식 (6)를 선택적으로 적용하여 각각 일반적 관절 경로 추종과 방역 로봇의 뉘기 동작을 구현했다.

5. 실험 및 결과

제안된 시스템의 유효성을 검증하기 위해, 장애물 치우기 및 닦기 작업의 계획에 소요되는 시간에 대한 분석과 모의 환경에서의 실험이 수행되었다. 실험 환경은 2.2에서 정의된 바와 같으며, 3.1에서 정의된 것과 같이 방역 대상면을 3 개의 구획으로 나누어 치우고 닦기 작업을 3 회 수행해 전체 대상면을 닦아냈다. 사용된 5×5×5 cm³ 정육면체와 10×5×5 cm³ 직육면체 장애물은 아래에서 각각 A 장애물과 B 장애물로 칭하도록 한다. 실험의 동작 생성에는 i9-10980XE CPU를 탑재한 컴퓨터가 사용되었으며, 동작 계획에는 Open motion planning library (OMPL)의 구현이 사용되었고^[20], 비구속 동작 계획은 1 초, 구속 동작 계획의 2초 이내에 해가 나오지 않을 경우 불가능한 동작으로 판단했다. [Fig. 3]에 요약된 전반적 작업 수준 계획 알고리즘은 Python을 통해 구현되었으며, 한 구획에 대한 작업 계획의 제한 시간은 300초로 설정했다. 실험 환경의 인식은 두 대의 RGBD 센서를 통해 이루어졌으나 (Azure Kinect, Intel Realsense), Depth 정보는 사용되지 않았고 각 요소는 ArUco^[21] 마커를 기반으로 인식되었다.

5.1 작업 및 동작 계획 시간

먼저, 제안된 동작 실현성 예측의 효과를 확인하기 위해 장애물의 수를 변경해가며 작업 계획을 10회씩 수행하고 실현성 예측 유무에 따른 계획 소요 시간을 측정해 정리했다[Table 2]. 장애물이 없는 상황에서는 실현성 예측 유무에 의한 차이가 없는데, 이는 정해진 구획을 순서대로 닦기만 하면 되기 때문에 동작 실현성에 대한 판단이 필요하지 않기 때문이다. 하지만, 장애물이 1 개 추가된 경우에는 과지 각도, 놓는 위치 등에 의해 불가능한 동작이 생기기 때문에 실현성 예측이 없는 경

[Table 2] Results of obstacle removal and surface wiping tests. Obstacle A is 5×5×5 cm³ cuboid and obstacle B is 10×5×5 cm³ cuboid. Better values in each scenario are marked in bold

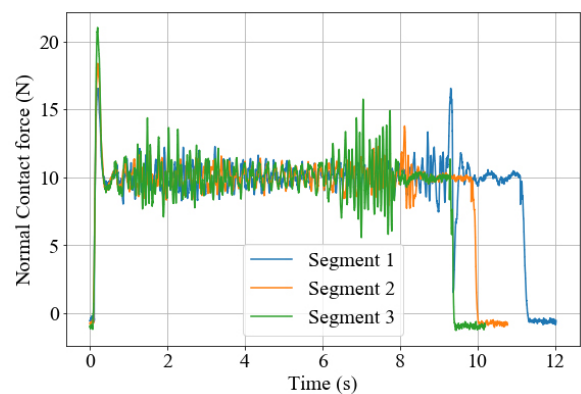
Obstacle count			Feasibility Prediction	Planning time (s)			Success rate (%)
Total	A	B		Min.	Mean	Max.	
0	0	0	X	1.24	2.41	5.48	100
			O	1.27	2.58	5.38	100
1	1	0	X	12.55	35.96	89.98	100
			O	1.96	5.57	9.93	100
2	1	1	X	200.97	300.14	414.09	40
			O	4.71	18.06	44.74	100
3	2	1	X	-	-	-	0
			O	8.06	41.55	72.76	100

우 평균 소요 시간이 약 15배로 급격히 늘어나 평균 35.96 초 가량이 소요되었다. 반면, 실현성 예측이 적용된 경우 평균 5.57 초 이내에 계획을 생성할 수 있었다. 또한, 실현성 예측 없이는 장애물이 2 개인 경우 300 초 이내에 한 구획의 계획에 실패하는 경우가 다수 발생했고 장애물 3 개인 경우 모든 경우에 실패했으나, 실현성 예측을 사용하는 경우에는 장애물이 3개인 경우에도 평균 41.55 초, 최대 72.76 초 이내에 전체 면에 대한 계획이 완료되었다.

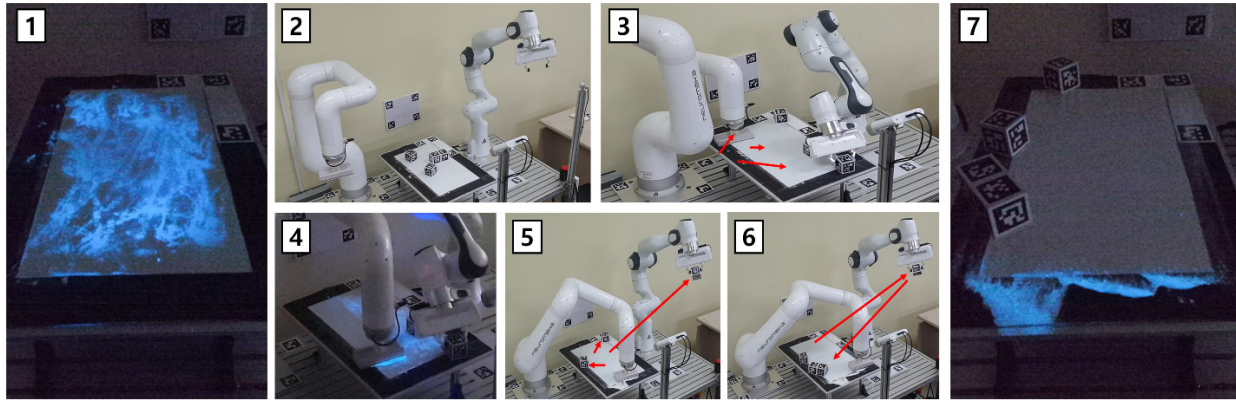
5.2 모의 환경의 방역 실험

앞선 과정으로 생성된 계획이 실제 방역 동작에 사용될 수 있는지를 검증하기 위해 표면에 도포된 형광 파우더를 제거하는 실험을 수행했다. 이 실험에서는 우선 방역 대상면에 형광 파우더를 도포하고 영상을 통해 형광 영역을 기록한 뒤, A 장애물 2 개와 B 장애물 1 개를 임의로 배치한 다음 치우고 닦기 작업을 수행했다. 작업 완료 후에는 다시 장애물을 치운 뒤 영상을 통해 대상 영역 [Fig. 1]의 검은 색 테두리를 제외한 흰색 영역) 상에 형광이 감지되는 영역을 다시 인식하고 작업 전/후에 형광 물질이 제거된 비율을 기록했다. 형광 물질 영역의 감지 기준은 Blue 채널과 Red 채널의 차이를 기준으로, 깨끗한 상태의 대상 영역 전체 중간 값의 두 배를 기준으로, 그보다 높은 값을 갖는 픽셀을 형광 영역으로 인식했다. 사진에서 드러나듯 형광 영역의 색상이 다른 곳과 뚜렷하게 구분되기 때문에, 이러한 기준으로도 충분히 구분이 이루어질 수 있었다.

이 실험과정에서 닦기 동작의 접촉력은 10 N으로 설정했으며, [Fig. 7]은 각 구획 별 닦기 동작 도중 측정된 접촉력을 나타낸 것이다. 각 동작의 도중에는 접촉력이 0 N 이하로 떨어진 순간이 없는 점을 통해 작업 중 대상면과의 접촉이 유지되고 있음을 확인할 수 있다.



[Fig. 7] Measured normal contact force while wiping each surface section



[Fig. 8] An example of obstacle removal and wiping process. (1) The target surface is polluted with fluorescent powder. (2) Obstacles are placed on the target surface. (3) Obstacles are removed from the first section. (4) The first section is wiped. (5) Obstacles are removed from the second section the second section is wiped. (6) Obstacles are removed from the third section and the third section is wiped. (7) The fluorescent powder is wiped out from the target surface. Red arrow shows movement of objects

[Table 3] Result of fluorescent powder removal over 10 trials

	Min.	Mean	Max.
Powder removal (%)	91.5	97.4	100.0

[Fig. 8]은 실험에서 수행된 치우고 닦기 과정의 진행 예시를 시간 순서로 표시한 것이다. [Table 3]는 이와 같은 과정으로 시행된 모의 방역 실험 10 회의 결과가 정리되어 있다. 최종적으로 평면에 도포된 형광 물질을 제거한 비율은 평균적으로 97.4%를 달성했으며, 실험 중 제거율이 가장 낮은 경우에도 91.5%가 제거되어, 제안된 시스템이 실제 방역 환경에서의 표면 방역에 적용될 가능성이 있음을 확인할 수 있었다.

6. 결 론

본 논문에서는 병원체 방역 과정 중 표면 소독에 필수적인 장애물 치우기 및 닦기 작업을 위한 작업 및 동작 계획 알고리즘을 제안했다. 제안된 작업 및 동작 계획 알고리즘은 장애물의 파지 자세 및 이동 위치 등의 작업 변수가 사전 설정되거나 이산화 되어 있지 않아도 계획 과정에서 스스로 결정될 수 있도록 했다. 또한, 다양한 경우의 수를 빠르게 검토하기 위해 충돌 검사와 SVM 기반 역기구학 및 자기충돌 예측 모델로 구성된 동작 실현성 예측 알고리즘을 제안했으며, 복수의 장애물이 있는 상황에서의 계획 속도와 성공률이 크게 향상됨을 보였다. 마지막으로, 계층적 제어를 통해 접촉력을 유지하는 닦기 동작을 구현하고 모의 환경에서 치우기 및 닦기 실험을 수행해 제안된 시스템의 실효성을 검증하였다.

References

- [1] D. B. Lee, "Post-Corona Era: The Present and Future of Antisaphtic Robots," *Convergence Research Policy Center*, Seoul, Korea, [Online], <https://crpc.kist.re.kr/user/nd49151.do?View&boardNo=00007243>.
- [2] "Coronavirus-19 Response Group Facility, Multi-Use Facility Disinfection Guide", *KDCA*, Sejong, Korea, [Online], <http://ncov.mohw.go.kr/shBoardView.do?brdId=2&brdGubun=25&ncvContSeq=3411>.
- [3] "Cleaning and disinfection of environmental surfaces in the context of COVID-19", *World Health Organization*, Geneva, Swiss, [Online]. <https://www.who.int/publications/i/item/cleaning-and-disinfection-of-environmental-surfaces-in-the-context-of-covid-19>.
- [4] L. P. Kaelbling and T. Lozano-Pérez, "Hierarchical task and motion planning in the now," *2011 IEEE International Conference on Robotics and Automation*, Shanghai, China, pp. 1470-1477, 2011, DOI: 10.1109/ICRA.2011.5980391.
- [5] S. Srivastava, E. Fang, L. Riano, R. Chitnis, S. Russell, and P. Abbeel, "Combined task and motion planning through an extensible planner-independent interface layer," *2014 IEEE International Conference on Robotics and Automation (ICRA)*, Hong Kong, China, pp. 639-646, 2014, DOI: 10.1109/ICRA.2014.6906922.
- [6] N. T. Dantam, Z. K. Kingston, S. Chaudhuri, and L. E. Kavraki, "An incremental constraint-based framework for task and motion planning," *The International Journal of Robotics Research*, vol. 37, no. 10, pp. 1134-1151, 2018, DOI: 10.1177/0278364918761570.
- [7] P. S. Schmitt, F. Wirschofer, K. M. Wurm, G. von Wichert, and W. Burgard, "Planning reactive manipulation in dynamic environments," *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Macau, China, pp. 136-143, 2019, DOI: 10.1109/IROS40897.2019.8968452.
- [8] D. Driess, J.-S. Ha, and M. Toussaint, "Deep visual reasoning: Learning to predict action sequences for task and motion planning from an initial scene image," *arXiv preprint arXiv:2006.05398*, 2020, [Online], <https://arxiv.org/abs/2006.05398>.

[9] A. M. Wells, N. T. Dantam, A. Shrivastava, and L. E. Kavraki, "Learning feasibility for task and motion planning in tabletop environments," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1255-1262, 2019, DOI: 10.1109/LRA.2019.2894861.

[10] D. Driess, O. Oguz, J.-S. Ha, and M. Toussaint, "Deep visual heuristics: Learning feasibility of mixed-integer programs for manipulation planning," *2020 IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France, pp. 9563-9569, 2020, DOI: 10.1109/ICRA40945.2020.9197291.

[11] J. Mirabel and F. Lamiroux, "Manipulation planning: addressing the crossed foliation issue," *2017 IEEE International Conference on Robotics and Automation (ICRA)*, Singapore, pp. 4032-4037, 2017, DOI: 10.1109/ICRA.2017.7989462.

[12] K. Hauser and V. Ng-Thow-Hing, "Randomized multi-modal motion planning for a humanoid robot manipulation task," *The International Journal of Robotics Research*, vol. 30, no. 6, pp. 678-698, 2011, DOI: 10.1177/0278364910386985.

[13] J. J. Kuffner and S. M. LaValle, "RRT-connect: An efficient approach to single-query path planning," *2000 IEEE International Conference on Robotics and Automation*, San Francisco, USA, pp. 995-1001, 2000, DOI: 10.1109/ROBOT.2000.844730.

[14] Z. Kingston, M. Moll, and L. E. Kavraki, "Exploring implicit spaces for constrained sampling-based planning," *The International Journal of Robotics Research*, vol. 38, no.10-11, pp. 1151-1178, 2019, DOI: 10.1177/0278364919868530.

[15] I. Rodriguez, K. Nottensteiner, D. Leidner, M. Kaßecker, F. Stulp, and A. Albu-Schäffer, "Iteratively refined feasibility checks in robotic assembly sequence planning," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1416-1423, 2019, DOI: 10.1109/LRA.2019.2895845.

[16] E. G. Gilbert, D. W. Johnson, and S. S. Keerthi, "A fast procedure for computing the distance between complex objects in three-dimensional space," *IEEE Journal on Robotics and Automation*, vol. 4, no. 2, pp.193-203, 1988, DOI: 10.1109/56.2083.

[17] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825-2830, 2011, [Online], <https://www.jmlr.org/papers/v12/pedregosa11a.html>.

[18] M. J. Kim, Y. Choi, and W. K. Chung, "Bringing Nonlinear H^∞ Optimality to Robot Controllers," *IEEE Transactions on Robotics*, vol. 31, no. 3, pp. 682-698, 2015, DOI: 10.1109/TRO.2015.2419871.

[19] A. Dietrich, C. Ott, and A. Albu-Schäffer, "Multi-objective compliance control of redundant manipulators: Hierarchy, control, and stability," *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Tokyo, Japan, pp. 3043-3050, 2013, DOI: 10.1109/IROS.2013.6696787.

[20] I. A. Sucas, M. Moll, and L. E. Kavraki, "The open motion planning library," *IEEE Robotics & Automation Magazine*, vol. 19, no. 4, pp. 72-82, 2012, DOI: 10.1109/MRA.2012.2205651.

[21] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognition*, vol. 47, no. 6, pp. 2280-2292, 2014, DOI: 10.1016/j.patcog.2014.01.005.



강준수

2014 포항공과대학교 기계공학과(학사)
2014~현재 포항공과대학교 기계공학과 석박사통합과정

관심분야: 작업 동작 계획, 로봇 비전, 로봇 학습



이인제

2020 영남대학교 기계공학과(학사)
2020~현재 포항공과대학교 기계공학과 석사과정

관심분야: 유압 로봇 제어, 매니플레이터 제어



정완균

1981 서울대학교 기계설계학과(학사)
1983 한국과학기술원 기계공학(석사)
1987 한국과학기술원 로봇공학(박사)
1987~1993 포항공과대학교 기계공학과 조교수
1993~2000 포항공과대학교 기계공학과 부교수
2000~현재 포항공과대학교 기계공학과 교수

관심분야: Robotic Manipulation, Polymer-based sensor and actuators, Visual recognition



김기훈

1999 포항공과대학교 기계공학과(학사)
2001 포항공과대학교 기계공학과(석사)
2006 포항공과대학교 기계공학과(박사)
2015~2019 한국과학기술연구원 책임연구원
2019~현재 포항공과대학교 기계공학과 부교수

관심분야: 바이오-의료 로봇, 재활로봇, 바이오닉스, 햅틱스