슬프게 걸어가는 휴머노이드 로봇

김 종 욱
동아대학교 전자공학과

Ⅰ. 서 론

휴머노이드 로봇은 일반적으로 20~45개의 액추에이터(모터)로 구성되어 있는 복잡한 구조의 로봇 플랫폼이다. 대학교의 로봇공학 교재들[1-2]이 최대 6자유도의 로봇 팔(manipulator)이나 2개의 바퀴가 달린 모바일 로봇을 대상으로 기구학(kinematics)과 동역학(dynamics) 등의 기본 이론을 설명하는 것을 생각하면, 이 정도의 높은 자유도를 이용해서 보행이나 동작을 구현하는 것은 결코 만만치 않음을 알 수 있다.

휴머노이드 로봇은 인간처럼 직립한 상태에서 움직이는 경우가 많으며, 이 상황에서 로봇을 움직이는 데 있어서 가장 큰 어려움은 바로 취약한 자세 안정성에 있다. 특히 이족보행(biped walking)을 위해서 로봇이 한쪽 발을 들 경우, zero moment point(ZMP, 영모먼트점)가 지지하는 발바닥 안에 위치해야 안정성을 확보할 수 있는데, 이는 여전히 복잡하고 어려운 기술이다.

본 고에서는 휴머노이드 로봇의 정서적 보행을 구현하는 과정을 단계적으로 요약하여 설명함으로써 휴머노이드 로봇 교육의 새로운 패러다임을 제시한다.

또한 수 천만원에서 수 억원씩 하는 휴머노이드 로봇으로 모션이나 보행 실험을 하다가 무리한 동작으로 팔이나 다리가 파손되거나 고장나는 경우가 발생하는 것을 미연에 방지하기 위해 모든 모션을 적용하기 전에 반드시 컴퓨터에서 시뮬레이션을 해 봐야 한다. 2015년 6월에 개최 된 DARPA Robotics Challenge(DRC) 결승전에서 대부분의 참가팀이 로봇용 메타 운영체제인 ROS(Robot Operating System) 기반에서 Gazebo 시뮬레이터로 로봇의 움직임을 3차원 그래픽으로 시뮬레이

션하면서 각종 미션을 수행한 것이 바로 그런 이유 때문이다.

본고에서는 로보티즈 사의 최신 휴머노이드 로봇 플랫폼인 로보티즈-OP(이하 OP)의 정서적 보행을 구현하는 과정을 [3]에서 소개된 내용을 요약하여 단계적으로 설명함으로써 휴머노이드 로봇 교육의 새로운 패러다임을 제시하고자 한다. 이를 위한 예로서 본고에서는 이족보행의 새로운 시도인 '휴머노이드 로봇이 슬픈 상태에서 걸어가는 것'을 구현하고자 한다.

2장에서는 최적화 알고리즘을 사용하여 일반적 보행 패턴을 시뮬레이션 하는 것을 설명하고, 3장에서는 OP의 미들웨어인 optWalk의 구조와 반사신경제어를 설명하며, 보행에 정서를 가미하기 위해 휴머노이드 로봇의 보행 속도, 보폭, 상체각, 눈의 LED 색깔 조정 등의 파라미터를 조절하는 원리를 설명하고 OP로 정서적 보행을 구현한 결과를 보인다. 5장에서는 결론 및 향후 연구 방향을 제시하며 끝을 맺는다.

DH법은 링크 좌표계 간의 기하학적 관계를 수학적으로 표현하는 방법이며, 휴머노이드 로봇을 모델링하는데 필수적이다.

Ⅱ. 최적 이족보행 시뮬레이션

본 장에서는 최적 이족보행을 위한 관절의 궤적을 자동으로 생성하는 과정을 설명하고자 한다. 이를 위해 필요한 지식은 기구학적 모델링, ZMP 개념, 관절 궤적 생성, 최적화 알고리즘이다.

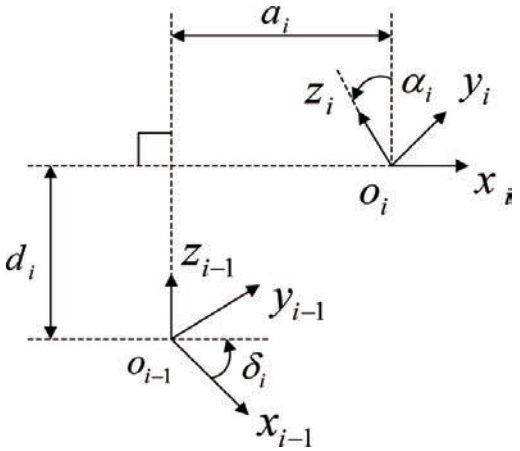
1. 기구학-로봇은 어떻게 표현되는가?

앞서 말했듯이, 이족보행을 포함한 전신 모션을 만들기 위해서는 시뮬레이터가 필요한데 이를 위해 기본적으로 필요한 것이 로봇을 수학적으로 모델링하는 것이다. 모델링의 첫 단계는 로봇의 링크와 관절을 정하는 것인데, 링크는 인체의 뼈처럼 로봇 전신의 형상을 튼튼하게 유지하는 역할을 하고, 관절은 링크를 회전시키거나(회전관절), 전후로 이동(병진관절)시킴으로써 로봇이 움직이게 하는 역할을 한다.

로봇의 링크와 관절의 숫자와 규격이 정해지면 각 링크에 지역(local) 직교좌표계를 할당해야 한다. 만약 A라는 사람이 거대한 휴머노이드 로봇의 어깨에 서 있고, B라는 사람은 그 로봇의 왼팔에 붙어 있다고 가정하자. 이 로봇이 왼팔을 앞뒤로 흔들때 A가 B를 보면 왼팔과 함께 B의 좌표와 방향이 바뀌는 것을 볼 것이고, 거꾸로 B가 A를 보면 A의 좌표와 방향이 반대 방향으로 바뀔 것이다. 하지만 B가 로봇의 왼손을 본다면 왼팔에 붙어 있기 때문에 로봇이 아무리 왼팔을 흔들어도 왼손의 좌표와 방향은 동일한 것처럼 보일 것이다. 이처럼 모든 링크는 상대적인 지역좌표계를 가지고 있으며, 링크별 지역좌표계의 x,y,z 축과 원점을 정하는 방식은 사람마다 다를 수 있다. 이 문제를 해결하기 위해 고안된 것이 바로 Denavit- Hartenberg(DH)법이다[5].

〈그림 1〉은 로봇공학 교재에서 반드시 소개되는 DH법의 두 인접한 좌표계($x_{i-1}y_{i-1}z_{i-1}$, $x_iy_iz_i$)간의 기하학적 관계도로서, δ_i는 $i-1$번째 좌표계를 z_{i-1}축을 중심으로 회전시킨 각도(관절각), d_i는 이렇게 변환된 좌표계를 다시 z_{i-1}축을 따라 이동시킨 거리(링크 오프셋), a_i는 이렇게 변환된 좌표계의 x_i축을 따라 좌표계를 이동시킨 거리(링크 길이), 마지막으로 α_i는 x_i축을 중심으로 좌표계를 회전시킨 각도(링크 비틀림)를 의미한다. 그리고 두 좌표계 간의 변환관계는 4×4 행렬로 표현된다[2].

〈그림 1〉과 같이 DH법은 두 인접한 링크에 고정되어



〈그림 1〉 DH법을 이용한 좌표계 간의 기하학적 관계도

있는 좌표계 간의 기하학적 관계를 표현한 것이므로, 그 다음 링크의 좌표계에도 이 방법을 순차적으로 적용하면 아무리 자유도가 높은 다관절 로봇이라 하더라도 기준좌표계(지지하는 다리의 발에 위치)로부터 다리와 팔, 머리의 말단부까지 모두 표현할 수 있다. 〈그림 2〉는 왼발로 지지(기준좌표계가 왼발에 위치)하고 있는 휴머노이드 로봇의 하체를 DH법을 따라서 정의한 것으로, 시상면(sagittal plane, 로봇을 옆에서 본 평면) 관절변수는 θ_1~θ_6, 관상면(coronal plane, 로봇을 앞에서 본 평면) 관절변수는 ϕ_1~ϕ_4, 횡평면(transversal plane, 로봇을 위에서 본 평면) 관절변수는 ψ_1~ψ_2이다. 그런데 발을 바꿔서 오른발로 지지하는 경우에는 기준좌표계가 오른발에 위치하므로 세 평면의 관절변수의 위치가 서로 바뀌게 된다. 각 좌표계의 DH 파라미터와 자세한 내용은 [3]을 참고하기 바란다.

지금까지 12자유도의 휴머노이드 로봇 하체를 DH법을 이용해서 모델링하는 과정을 설명했다. 상체도 동일한 원리로 각 링크별로 지역좌표계를 구현한 후, 전역(global) 직교좌표계에서 통합해서 표현하면 된다.

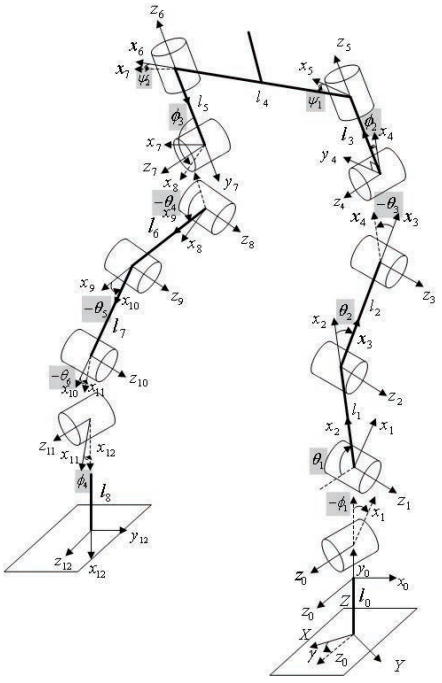
로봇의 자세 안정도를 측정하기 위해 지지하는 다리의 발바닥이 만드는 다각형 안에 ZMP가 존재하는지를 확인해야 한다.

2. ZMP–로봇의 자세 안정도

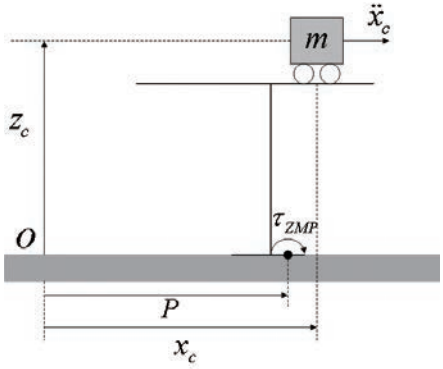
〈그림 2〉에 나타낸 하체의 12개 관절과 상체 8개 관절에 특정 각도 값을 부여하면 그에 따라 휴머노이드 로봇은 하나의 정지 자세를 취할 것이다. 그리고 그 정지 자세를 제어기에서 샘플 타임 간격으로 조금씩 조정해서 전송한다면 로봇은 움직이게 된다.

휴머노이드 로봇의 발바닥은 바퀴나 무한궤도에 비해 상대적 면적이 작은 편이므로, 로봇이 움직이는 동안 자세 안정도를 유지하기가 상당히 어렵다. 이를 위해서는 자세 안정도를 측정하는 방법이 필요한데, Vukobratović 은 1968년에 ZMP 이론을 발표했다[6]. 이 이론에 따르면 로봇의 모든 관절이 발생시키는 힘의 총합을 한 지점에서 발생하는 단일 힘으로 표현한 후, 이 힘이 영이 되게 하는 점의 위치를 계산한다. 이 점을 ZMP라 하며, 지지하는 발(한 발 또는 양 발)의 바닥이 만드는 다각형 안에 ZMP가 있으면 이 로봇은 그 순간 안정하다는 것이다.

〈그림 3〉은 ZMP를 이해하기 쉽게 설명한 카트–테이블 모델로서 카트는 휴머노이드 로봇의 무게 중심점, 테이블의 지지대는 로봇을 지지하는 발을 의미한다. 카트가 테이블의 가운데에 위치하면 안정적이지만, 테이블의 지지대를 벗어나서 앞 쪽에 위치하면 중력에 의해 발생된 토크($mg(x_c-P)$)에 의해 테이블은 앞으로 넘어질 것이다.



〈그림 2〉 휴머노이드 로봇 하체의 DH법 모델링



〈그림 3〉 ZMP를 설명하기 위한 카트–테이블 모델

이 순간 카트가 $\ddot{x}_c$의 가속도로 전진하면 어떤 일이 발생할까? 바로 '작용-반작용의 원리'에 의해 카트가 발생시킨 힘만큼 테이블이 뒤로 밀리게 되는데 바닥의 마찰력 때문에 그만큼의 회전력, 즉 역방향의 토크($-m\ddot{x}_c z_c$)가 발생하게 된다. 이 두 가지 상반되는 토크의 합을 계산해서 그 합이 0이 되게 하는 점 P가 바로 ZMP이다. 결과적으로 카트의 전진 때문에 ZMP가 테이블 지지대 안쪽으로 들어오게 되어 테이블이 앞으로 넘어지는 것을 방지한다.

그런데, 실제 휴머노이드 로봇은 분리되어 있지 않은데, 카트와 테이블 모델에서는 서로 분리되어 있어서 의아할 것이다. 이 모델에서 카트는 로봇의 전신을 지지하는 다리를 나타낸다면 테이블은 그 이외의 부분(들어 올린 다리, 상체와 팔, 머리를 모두 포함한 것)을 나타낸다. 그러므로 〈그림 3〉에서 카트가 앞으로 나가는 것은 들어 올린 다리를 앞으로 내뻗는 동작이나 상체를 앞으로 굽히는 동작을 의미한다. 그러므로 이족보행에서 전진하기 위해 한쪽 발을 들어 올리는 것은 카트가 앞으로 쏠리는 상황에 비유될 수 있고, 들어 올린 발을 앞으로 뻗는 것은 카트를 전진시키는 것과 같은 역할을 함으로써 자세 안정도를 강화하는 역할을 하는 것이다.

이러한 이유로 휴머노이드 로봇이 움직이는 동안 매 순간의 ZMP를 계산하여, 지지하는 다리의 발바닥이 만드는 다각형 안에 최대한 존재하는 지를 확인하는 것이 자세 안정도를 위해 필요하다.

3. 관절궤적 생성-로봇 움직이기

이제 로봇을 움직여 보도록 하자. 사람과 마찬가지로 구현하려는 모션과 관련된 관절을 적절히 부드럽게 회전시켜야 하고, 이 모든 관절 회전의 조합으로 로봇은 보행하고 움직이게 된다.

관절을 회전시키기 위해서는 각 샘플 시간마다 회전각을 명령값으로 보내야 한다. 한두 개의 모터라면 일일이 회전각을 정해줄 수 있지만, OP만 해도 모터가 20개에 달하므로 각 모터에 대해 이렇게 하기에는 대단히 어렵다. 게다가 샘플 시간이 바뀌거나 특정 모터의 회전각을 증가 또는 감소시켜야 하는 경우 이 과정은 더욱 복잡해진다. 이러한 문제를 해결하는 방법은 각 관절에 몇 개의 파라미터로 조정할 수 있는 부드러운 다항함수 궤적을 할당하는 것이다. 특히, 휴머노이드 로봇의 경우 배터리를 사용하기 때문에 움직일 때 전력소비를 최소화해야 하는데, 이를 위해서 관절모터는 최대한 부드럽게 회전해야 한다.

부드러운 로봇 모션을 용이하게 만들기 위해, 저차의 다항함수를 이어 붙인 혼합 다항함수를 이용하여 관절의 각도각 궤적을 생성한다.

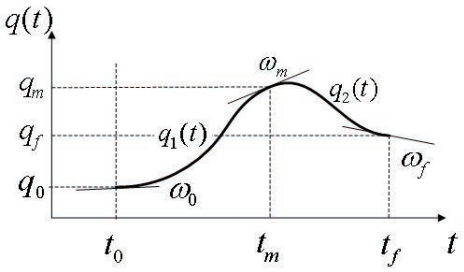
일반적으로 한 주기 동안의 관절각 궤적에 사용되는 다항함수는 최고차수가 3차에서 5차에 이르는데[2], 차수가 높아질수록 관절의 회전각 궤적의 모양은 복잡해진다. 이렇게 되면 최고차항의 계수에 따라 회전각이 큰 값으로 발산할 수 있는 문제점이 있다. 그러므로 저차의 다항함수를 유지하면서 다양한 궤적을 만들 수 있는 혼합 다항함수(blending polynomial)를 사용하는 것이 유리하다[2].

혼합 다항함수는 전체의 궤적을 원하는 구간으로 나누고 각 구간을 다음과 같이 3차의 다항함수로 표현한 다항함수이다.

$$q(t) = a_0 + a_1(t-t_0^i) + a_2(t-t_0^i)^2 + a_3(t-t_0^i)^3 \qquad t_0^i \le t \le t_f^i \tag{1}$$

여기서 $a_0, \cdots, a_3$는 궤적의 계수인데, 다음과 같이 i번째 구간의 시작시간 t_0^i와 마침시간 t_f^i의 각도와 각속도 조건으로 표현된다.

$$a_0 = q_0^i,\ a_1 = \omega_0^i,$$
$$a_2 = \frac{3(q_f^i - q_0^i) - (2\omega_0^i + \omega_f^i)(t_f^i - t_0^i)}{(t_f^i - t_0^i)^2}$$
$$a_3 = \frac{2(q_f^i - q_0^i) + (\omega_0^i + \omega_f^i)(t_f^i - t_0^i)}{(t_f^i - t_0^i)^3} \tag{2}$$



〈그림 4〉 2개의 구간을 가진 혼합다항함수로 표현된 회전각 궤적

식에서 q_0^i, q_f^i와 ω_0^i, ω_f^i는 시작시간과 마침시간에서의 각도와 각속도를 각각 나타낸다. 이 네 값을 정해주면 식 (2)에서 네 개의 계수를 자동으로 계산하고, 이 값으로부터 식 (1)을 이용하여 i번째 구간의 회전각 궤적을 다항함수로 그릴 수 있다. 〈그림 4〉는 2개의 구간을 가진 다항함수 궤적을 예시한 것이다.

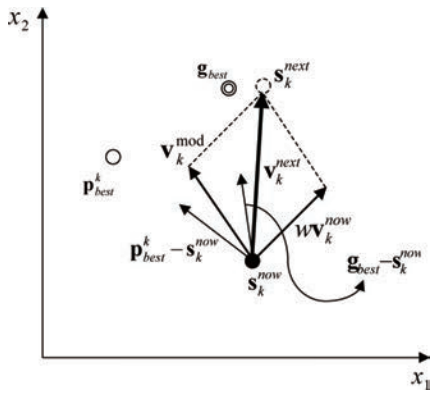
4. 최적화 알고리즘으로 보행패턴 자동 생성

지금까지의 내용을 잘 이해했다면 한 가지 의문이 생길 것이다.

"휴머노이드 로봇 모델도 구축했고, 자세 안정도를 나타내는 값으로 ZMP를 사용할 수 있으며, 각 모터에 보낼 각도 명령을 혼합다항함수로 표현하는 방법도 배웠다. 그런데, 보행을 위해 관절 모터에 명령값으로 보낼 각도 값은 어떻게 정할 수 있는가?"

사실, 로봇이 원하는 보폭과 속도로 사람처럼 안정적으로 걷기 위해서는 다음의 부가적인 조건들을 모두 만족시킬 필요가 있다.

- 보행 중에 ZMP가 지지하는 발의 뒤꿈치에서 앞쪽으로 자연스럽게 이동해야 하며, 양측지지 시에는 두 발의 중심점으로 다시 돌아와야 한다.
- 머리에 있는 카메라의 흔들림을 최소화하기 위해 상체는 지면과 가급적 90°를 이뤄야 한다.



〈그림 5〉 2차원 탐색공간에서 PSO의 k번째 개체의 다음 차수에서의 위치와 방향

- 들어 올린 다리는 지지하는 다리와 평행을 유지해야 한다.
- 들어 올린 발의 바닥면은 지면과 평행해야 하며 가급적 정규분포와 같은 궤적을 그려야 한다.
- 자기 충돌(self collision)을 피하기 위해 상체의 두 팔은 관상면에서 항상 15° 정도 바깥쪽으로 벌려져 있어야 한다.

만약 여러분이 OP의 자체 모션 편집기인 액션에디터를 이용해서 앞의 조건들을 모두 만족시키는 보행 패턴을 만들고자 한다면 상당히 많은 시행착오를 거쳐야 한다. 게다가 로봇 모델이 바뀌기라도 한다면 그 작업을 처음부터 다시 해야 하는 것이다.

다행히 최적화 알고리즘(optimization method)을 사용하면 이 모든 조건을 골고루 만족시켜주는 해를 컴퓨터 프로그램을 이용해서 얻을 수 있다. 수많은 최적화 알고리즘 중에서 가장 간단하면서도 만족할 만한 탐색 성능을 보이는 것은 particle swarm optimization(PSO)[7]이며, 〈그림 5〉는 2차원 탐색 공간에서 PSO의 탐색 원리를 나타낸다.

최적화 알고리즘인 PSO를 사용하여 주요 관절궤적 파라미터를 최적화함으로써 안정된 이족보행에 필요한 기본 조건을 모두 만족시키는 보행패턴을 만들 수 있다.

현재 개체의 2차원 위치 벡터를 s_k^{now}라고 하고 이 개체가 지금까지 탐색한 해 중 가장 좋은 해를 p_{best}^k, 집단 전체가 현재까지 탐색한 해 중 가장 좋은 해를 g_{best}라고 한다. 그러면 현재 개체의 다음 반복에서의 위치 벡터인 s_k^{next}는 p_{best}^k와 g_{best}쪽을 향하고 직전 속도를 가급적 유지하면서, 새로운 해를 찾기 위해 무작위성이 추가되어야 한다. 이를 식으로 표현하면 다음과 같다.

$$s_k^{next} = s_k^{now} + v_k^{next},\ k = 1, \cdots, N$$
$$v_k^{next} = wv_k^{now} + c_1 r_1 \left(p_{best}^k - s_k^{now}\right) + c_2 r_2 \left(g_{best} - s_k^{now}\right) \quad (3)$$

식에서 N은 PSO 집단의 전체 개체수를 나타내며, ω, c_1, c_2는 가중치 계수, r_1, r_2는 0과 1 사이에서 발생시킨 균일분포 랜덤 값을 나타낸다.

이제 PSO를 이용해서 앞의 5가지 조건을 모두 만족시키는 이족 보행 패턴을 혼합다항함수 기반으로 만들어 보

자. 그런데, PSO의 변수의 수가 증가할수록 최적화 시간은 비례하고 최적화 성능은 반비례하기 때문에 꼭 필요한 최적화 변수만을 엄선해서 최적 보행을 위한 관절 궤적 패턴을 생성해야 한다. 이를 위해 다음의 접근법을 사용했다.

- 왼발로 지지하는 상황을 기준으로 최적화하며, 보행 시 오른발로 지지할 때에는 하체 관절의 각도 궤적을 서로 바꾸면 됨
- 한 스텝의 시작과 끝의 자세를 원하는 보폭에 맞춰 만들어 놓음으로써 각 관절궤적의 초기각과 최종각을 미리 계산 함
- 6개의 하체 시상면 관절각(θ_1~θ_6) 중 상체각을 조정하는 θ_3와 들어 올린 발의 각도를 조정하는 θ_6는 나머지 관절각이 변하더라도 상기 보행조건을 만족시키도록 자동으로 계산 함. 나머지 관절각은 2구간 혼합다항식으로 궤적 표현
- 4개의 하체 관상면 관절각(ϕ_1~ϕ_4) 중 상체각을 조정하는 ϕ_2와 들어 올린 발의 각도를 조정하는 ϕ_4는 나머지 관절각이 변하더라도 상기 보행조건을 만족시키도록 자동으로 계산 함. 나머지 관절각은 3구간 혼합다항식으로 궤적 표현

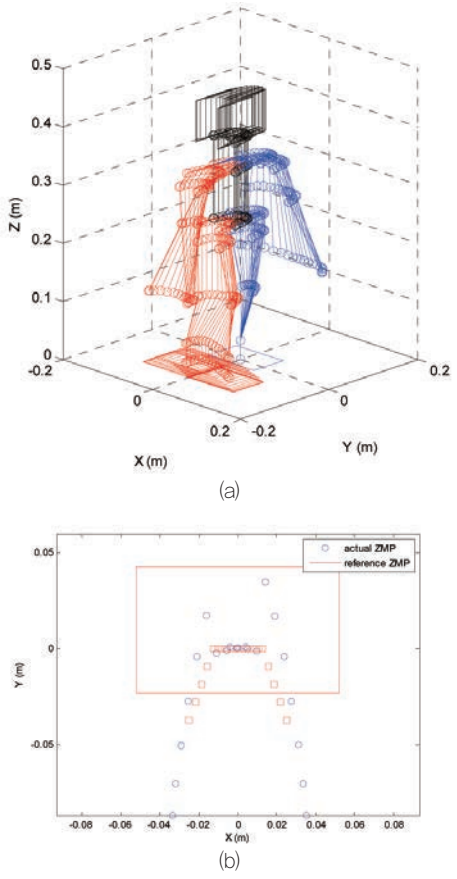
저자는 시뮬레이터를 통해 검증한 최적 이족보행 패턴을 로봇 플랫폼으로 구현하기 위해 MATLAB 코드와 호환되는 optWalk 미들웨어 패키지를 C++로 개발했다.

이 조건들을 모두 적용하고 전신의 대칭조건을 적극적으로 활용하면 PSO로 최적화할 변수는 다음과 같이 13개가 된다.

$$\Theta = [\theta_{an}^{l,M},\ \theta_{kn}^{l,M},\ \theta_{hp}^{r,M},\ \theta_{kn}^{r,M},\ \omega_{an}^{l,M},\ \omega_{kn}^{l,M},\ \omega_{hp}^{r,M},\ \omega_{kn}^{r,M},\ t_{M1},\ t_{M2},\ \left|\phi_{an}^{\max}\right|,\ \theta_{sh}^{l,M},\ \omega_{sh}^{l,M}] \quad (4)$$

식에서 θ와 ω는 시상면 관절의 각도와 각속도를, ϕ는 관상면 관절의 각도를 나타낸다. 아래첨자 *an*, *kn*, *hp*, *sh*는 발목관절, 무릎관절, 고관절, 견관절을 나타내고, 위첨자에서 *l*,*r*은 왼쪽과 오른쪽을 나타내며, *M*은 중간값을 나타낸다.

〈그림 6〉은 앞 페이지에서 언급한 보행 조건을 모두 만족시키도록 PSO로 최적화한 변수를 이용해서 이족보행을 MATLAB으로 시뮬레이션 한 것이다. 〈그림 6(a)〉는 보폭 5cm, 주기 1초로 보행을 하는 모습을 샘플 시간별로 보인 것이며, 〈그림 6(b)〉는 목표로 한 ZMP 궤적을 잘 추종하는 것을 보인다.



〈그림 6〉 목표 ZMP 궤적을 추종하는 최적 이족보행 시뮬레이션

Ⅲ. OP로 정서적 보행 구현

본 장에서는 2장에서 마지막 부분에 PSO로 얻은 최적 관절 궤적을 OP에 적용하여, 안정되면서도 특징적으로 걸어가는 과정을 설명하고자 한다. 이를 위해 저자는 MATLAB 시뮬레이터의 소스코드와 호환되는 optWalk라는 미들웨어 패키지를 개발했다. 이 장에서는 먼저 optWalk 패키지의 구조를 설명하고, 반사신경 제어

(sensory reflex control)를 이용해서 현재 자세와 바닥 환경에 따라 관절궤적을 조절하는 방법, 그리고 OP로 정서적 보행을 구현하기 위해 보행 특징을 조정하는 방법에 대해 설명한다.

1. optWalk-로봇 내부의 모션 패키지

optWalk 패키지는 〈그림 7〉과 같이 C++ 코드로 작성된 파일로 구성되어 있으며, 각 파일의 역할은 다음과 같다.

- main.h/cpp: 텍스트 기반 UI를 이용한 전체 모션 흐름 제어
- MathFunc.h/cpp: 각종 수치연산에 필요한 수학함수 정의
- pso.h/cpp: 모션과 자세 최적화를 위한 PSO 수행
- JointAngles.h: OP의 모든 관절각 관련 파라미터 정의
- SelfMotion.h/cpp: 이족보행을 포함한 OP의 다양한 동작 생성 및 제어 함수 정의
- robotisOP.h/cpp: OP의 규격정보 저장 및 기구학/동역학 연산 수행
- MotionDB.h: OP의 다양한 동작과 관련된 관절변수와 관절궤적 데이터 저장

2장에서 행했던 MATLAB 시뮬레이션에서는 물리적 엔진을 사용하지 않았고 바닥은 완전한 평지라고 가정했기 때문에 넘어지지 않고 잘 걸어갔다. 그리고 ZMP를 계산할 때 휴머노이드 로봇의 각 링크별 무게중심점이 링크의 정중앙에 위치했다고 가정했지만 실제로는 그렇지 않다. 특히 상체의 경우 내부에 장착된 SBC(단일보드 컴퓨터)와 배터리의 무게 때문에 실제 무게중심점은 뒤편 아래쪽에 위치하므로, 시뮬레이션에서 안정한 궤적이라 하더라도 그대로 OP에 적용하여 걸어가게하면 뒤로 넘어지게 된다. 그러므로 스텝 시작 자세와 마침 자세를 약간 조정하고 상체각을 적절히 앞뒤로 움직여줘야 한다.

특히 휴머노이드 로봇이 약간의 경사가 있는 바닥을 걸어갈 때는 인간의 반사신경제어 기법을 각 관절에 적용해야 한다.

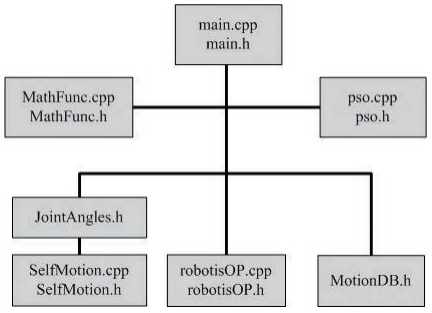
2. 반사신경제어

Huang 등은 인간의 리드미컬한 보행은 내적인 패턴과 반사적 행동을 적절히 조합한 결과라고 해석하고, 이러한 원리를 이용하여 휴머노이드 로봇의 안정된 보행을 위한 반사신경 제어(sensory reflex control) 기법을 제안했다[8]. 〈그림 8〉은 반사신경 제어기의 전체구조를 나타내며, θ_p는 제어기에 피드포워드(feedforward) 되는 동적인 내적패턴 값을 의미하고, $\Delta\theta(t)$는 로봇의 현재 자세로부터 측정된 센서 데이터를 기반으로 피드백(feedback)되는 관절각 조정 값을 의미한다. 그림에서 θ_r은 목표 관절에 인가되는 각도 명령이며 다음 식으로 표현된다.

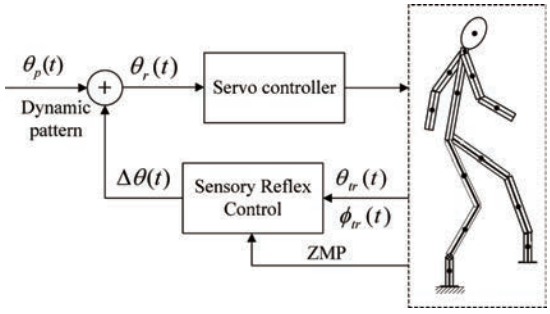
$$\theta_r(t) = \theta_p(t) + \Delta\theta(t) \tag{5}$$

휴머노이드 로봇의 경우 발바닥의 압력점 위치와 상체각을 측정하여 현재 자세의 안정도를 측정해야 하며, 이를 이용하여 ZMP와 상체각을 기반으로 하는 반사신경제어를 적용한다.

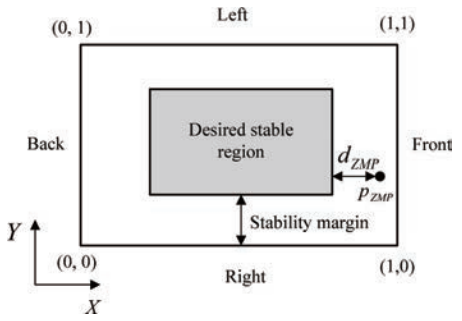
ZMP 반사신경제어는 〈그림 9〉의 안정도 측정값인



〈그림 7〉 optWalk 패키지의 구조



〈그림 8〉 반사신경 제어기의 구조



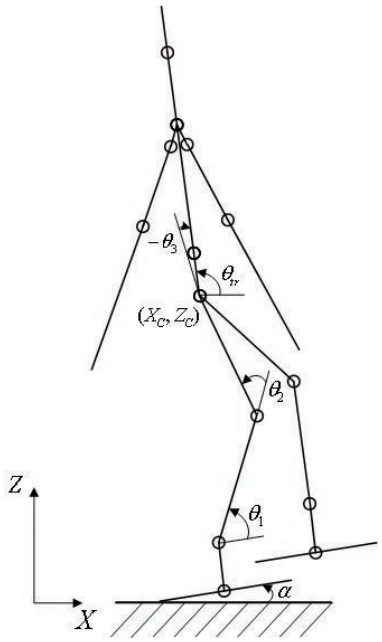
〈그림 9〉 지지하는 발바닥의 안정된 ZMP 영역과 안정도 측정값

d_{ZMP}를 이용해서 시상면 하체 지지다리의 족관절, 슬관절, 고관절을 조절한다. d_{ZMP}는 현재 ZMP(p_{ZMP})가 〈그림 9〉에 보인 목표안정영역(desired stable region)을 앞쪽으로 벗어나면 양수, 뒤쪽으로 벗어나면 음수가 되도록 정의했다.

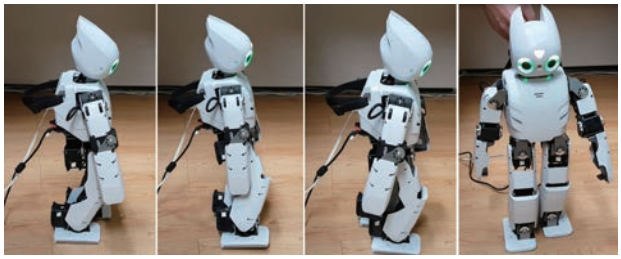
족관절 제어는 어떤 모션을 하다가 d_{ZMP}가 양수나 음수가 되는 즉시 족관절을 현재 각도값 이상으로 회전하지 못하도록 제한하는 원리이다. 슬관절 제어는 d_{ZMP}가 양일 때는 골반중심점을 그만큼 뒤로 이동하게 하고, d_{ZMP}가 음일 때는 골반중심점을 그만큼 앞으로 이동하게 함으로써 좀 더 빨리 ZMP가 안정영역 안으로 돌아오게 한다. 마지막으로 고관절 제어는 d_{ZMP}가 양일 때는 상체를 뒤로 회전시키고 음일 때는 앞으로 회전시킴으로써 전신의 자세 안정도를 빨리 회복시키는데 기여 한다.

상체각 반사신경 제어기는 좀 더 직관적이고 단순하다. 〈그림 10〉에서 보듯이 상체가 약간 뒤로 젖혀진 경우 상체의 시상면 관절각 θ_{tr}은 90° 보다 커진다. 이 경우 시상면 족관절각 θ_1을 현재보다 그만큼 더 감소시켜야 상체가 다시 직각이 된다. 마찬가지로 시상면 고관절각 θ_3 또한 그만큼 감소되어야 상체가 직각에 가까워진다.

로봇이 제자리에 두 발로 서서 움직이는 모션의 경우 FSR을 이용해서 d_{ZMP}를 측정하기가 용이하므로 ZMP 반사신경 제어를 적용할 수 있지만, 이족보행의 경우에는 그렇지 못하므로 상체각 반사신경제어를 적용하는 것이 훨씬 안정적이다. 그러나, 물건을 들고 걸어가는 경우에는 그 물건을 포함한 무게중심점이 가운데로 오기 위해 상체를 뒤로 약간 젖혀야 하므로 ZMP 반사신경제어를 활용할 필요가 있다.



〈그림 10〉 시상면에서의 상체각과 하체 관절각



〈그림 11〉 OP가 슬픈 상태에서 걸어가는 장면을 시상면과 관상면에서 찍은 스냅샷

마지막으로, OP를 이용해서 정서적 보행을 구현해 보도록 하자. 본 논문에서는 대표적인 정서적 보행으로 휴머노이드 로봇이 슬픈 감정 속에서 처량하게 걸어가는 것을 구현하고자 한다. 인체보행 실험을 통해 인간은 정상적인 상태에 비해 슬픈 경우 다음과 같은 특징이 나타남을 확인했다[9].

- 보행속도가 느려짐
- 보폭이 감소함
- 찡그린 표정
- 상체를 약간 앞으로 숙임

보행속도가 느려지고 보폭이 감소하는 것은 혼합다항

식 궤적의 각 경유점에서의 파라미터 세트(〈그림 4〉에서 보듯이 시간값, 각도값, 각속도값)를 비례 또는 반비례해서해서 조절하면 된다. 즉, 보행속도를 감소시키기 위해서는 시간값을 증가시키고, 각속도값은 그만큼 감소시켜야 한다. 그리고 보폭을 감소시키기 위해서는 각도값과 각속도값 모두를 그만큼 감소시켜야 한다.

찡그린 표정은 OP로 구현하기 어려우므로 얼굴의 LED를 녹색으로 바꾸는 것으로 표현했고, 상체를 약간 앞으로 숙이는 것은 〈그림 9〉의 θ_{tr}가 90° 이하가 되는 것이므로 θ_3를 적절히 감소시키면 된다. 〈그림 11〉은 OP가 이와 같은 방법으로 슬프게 걸어가는 것을 스냅샷으로 찍은 것이다.

Ⅵ. 향후 연구 및 결론

지금까지 저자가 집필한 휴머노이드 로봇 교재[3]의 내용을 기반으로 정서적 보행의 일례인 '슬프게 걸어가기'를 OP로 구현하는 과정을 소프트웨어(MATLAB) 시뮬레이션, 미들웨어(optWalk) 코딩, 하드웨어(OP) 구현의 순서로 설명했다. 본 교재에는 이족보행 뿐만 아니라 전신 모션과 LED 색깔 조정 등 다양한 기능들이 소스코드와 함께 소개되어 있으며, OP의 SBC를 기반으로 인공지능 기술 등의 다양한 프로그램을 작성해서 테스트해 볼 수 있으므로 현재 점차 강화되고 있는 소프트웨어 교육에도 적절히 사용될 수 있을 것이라고 기대한다.

참 고 문 헌

[1] K.S. Fu, R.C. Gonzalez, and C.S.G. Lee, Robotics: Control, Sensing, Vision, and Intelligence, McGraw-Hill, Inc., 1987.

[2] M. W. Spong, S. Hutchinson, and M. Vidyasagar, Robot Modeling and Control, John Wiley & Sons, Inc., 2006.

[3] 김종욱, 함께 즐기는 휴머노이드 로봇: 로보티즈 OP편, 홍릉과학출판사, 2015.

[4] Mathworks, http://kr.mathworks.com/

[5] J. Denavit and R. S. Hartenberg, "A kinematic notation for lower-pair mechanisms based on matrices," Journal of Applied Mechanics, vol. 77, pp. 215-221, 1955.

[6] M. Vukobratović and D. Juričić, "Contribution to the synthesis of biped gait," Proc. IFAC Symp. Technical and Biological Problem on Control, Erevan, USSR, 1968.

[7] J. Kennedy and R. Eberhart, "Particle swarm optimization," Proceedings of IEEE International Conf. on Neural Networks, vol. 4, pp. 1942-1948, Nov/Dec 1995.

[8] Q. Huang, Y. Nakamura, "Sensory reflex control for humanoid walking," IEEE Trans. Robotics, vol. 21, no. 5, Oct. 2005.

[9] 당 반 치엔, 트란 트렁 틴, 김종욱, "다양한 감정 상태에서의 보행 특징 분석", 한국지능시스템학회논문지, 24권, 5호, pp. 477-481, Oct. 2014.

김 종 욱

- 1997년 8월 포항공과대학교 전자전기공학과 학사
- 2000년 2월 포항공과대학교 전자전기공학과 석사
- 2004년 2월 포항공과대학교 전자전기공학과 박사
- 2004년 3월~2004년 5월 포항공과대학교 박사후 연구원
- 2004년 6월~2006년 2월 포스코 기술연구소 선임연구원
- 2006년 3월~현재 동아대학교 전자공학과 부교수

〈관심분야〉
휴머노이드 로봇, 최적화 알고리즘, 지능제어, 임베디드 시스템