

모듈라 로봇 이동 제어에 대한 진화연산 기법 비교

A Comparison Study of Evolutionary Computation Techniques for Locomotion Control of Modular Robots

장재영, 현수환, 서기성*
(Jae Young Jang, Soohwan Hyun and Kisung Seo)

Abstract : Modular Snake-like robots are robust for failure and have flexible locomotion for environments, but are difficult to control. Various phase and evolutionary approaches for modular robots have been studied for many years, but there are few comparisons among these methods. In this paper, Phase, Gaps, GA and GP approaches are implemented and compared for flat, stairs, and slope environments. In addition, modular Snake-like robot are simulated in Webots environments.

Keywords : Modular Robot, Snake-like, Locomotion, Genetic Algorithm, Genetic Programming

I. 서론

모듈라 로봇은 독립적인 단위 구성 요소인 모듈을 원하는 기능을 수행할 수 있도록 특정한 형태로 연결하여 만든다[1,2]. 구성 형태에 따라 뱀형, 거미형, 루프형, 보행형등 다양한 종류의 로봇 구현이 가능하다. 또한, 모듈의 특성상 쉽게 연결 구성을 바꿀 수 있으므로 재구성(reconfiguration)이 가능하며, 일부 모듈의 고장이나 결함에도 불구하고 상당한 강인성을 가지는 장점이 있다[1,2].

모듈라 뱀형 로봇은 다수의 모듈을 일렬로 연결하여 구성한 이동로봇으로서, 기존의 바퀴나 보행 로봇에 비하여 다음과 같은 장점을 가지고 있다. 일부분이 고장 났을 경우에 정상적인 모듈들만으로도 동작이 가능하며, 환경이 불안정 하거나, 장애물이 있는 지형에서도 유연하게 움직일 수 있는 특성을 갖는다. 그러나 기존의 바퀴나 보행 로봇에 비하여 이동속도가 느리고 제어가 어렵다는 단점이 있다[3,4].

기존의 모듈라 뱀형 로봇의 이동 제어에 사용된 방식으로는 삼각함수를 이용해 로봇 전체의 움직임을 표현한 후, 그것을 각각의 모듈에 대해 일정 주기만큼 수평이동 하여 지속적으로 신호를 입력하는 방식이 있다[5]. 그리고 CPG (Central Pattern Generator)를 이용해 이어가는 동작과 헤엄치는 동작을 생성한 방법 등이 있다[3]. 그러나 이러한 방법들로 이동 방법(locomotion)을 설계하는 방식은 최적화하는 거리가 멀고, 주로 생물학적 운동 형태를 모방하여 안정적인 움직임을 생성하는데 치우쳐 있다. 따라서, 속도와 같은 성능의 개선에는 미흡한 형편이다.

이에 반하여, 최근 들어 모듈라 로봇의 이동 제어에 진화연산 기법을[6,7] 사용한 연구가 수행되어 왔고, 괄목할 만한 성과를 얻고 있다[3,4,12]. 진화 연산을 이용할 경우,

수 없는 보다 다양한 이동 방식의 생성이 가능하다고 알려져 있다.

그러나, 이들 대부분의 연구에서는, 사용된 모듈이 가상적이거나[4,12], 진화연산과 일반적인 주기 방식의과의 비교만을 한것으로, 다양한 진화 접근 방법에 대한 비교 연구는 거의 이루어지지 않고 있다. 즉, ZMP와 COM 정보등을 이용하여 걸음새에 대한 체계적인 최적화 연구가 많이 이루어진 이족보행의 경우처럼 뚜렷한 비교 대상의 기법이 없으면, 진화연산 접근법의 유용성의 정도가 객관화 될수 있다. 또한, 모듈라 로봇의 경우는 아직까지 이동 방식에 대한 충분한 연구가 이루어지고 있지 않으므로, 주기적인 접근방식과의 단일 진화방식과의 비교만을 통한 연구는 객관성이 떨어질수 있다. 또한 진화연산의 여러기법도 각 기법의 특성상 다른 진화 기법에 비해서 모든 환경에 우수한 특성을 찾기란 매우 어렵다.

따라서, 본 연구는 먼저 다양한 환경에서의 모듈라 로봇의 이동 방식에 대해서 경쟁력있는 진화연산 방식들을 비교하고, 이들의 특성을 분석하여, 향후 좀 더 개선되고 정교한 탐색기법 구현을 위한 교두보를 마련하고자 한다.

이를 위하여 뱀형 모듈라 로봇의 이동 제어에 다음 3가지 방식을 비교하였다 - GA와 주기방식의 결합(GAps), GA 기반 방식, GP 기반 방식. 대상 로봇은 H/W 적으로 구현이 가능한 실제 모듈로 구성되었으며, 물리적 특성의 시뮬레이션이 가능한 Webots[8]을 사용하여 시뮬레이션 실험을 수행하였다.

II. 모듈라 뱀형 로봇의 이동 제어

1. 모듈라 뱀형 로봇 모델

본 논문의 실험에서 사용된 모듈은 BIRG(Biologically Inspired Robotics Group)의 모듈라 로봇의 자가 가구 구성 (Modular robotics for adaptive and self-organizing furniture)에서 사용되는 기본 모듈로써 명칭은 Roombot이다[9].

* 책임저자(Corresponding Author)

논문접수 : 2008. X. XX., 채택확정 : 2008. X. XX.

장재영 : 서경대학교 전자공학과(geruld@naver.com)

현수환 : 서경대학교 전자공학과(xjavalov@shhyun.com)

서기성 : 서경대학교 전자공학과(kseo@skuniv.ac.kr)

기존의 경험적이거나 주기적인 수식 접근법에서는 예측할

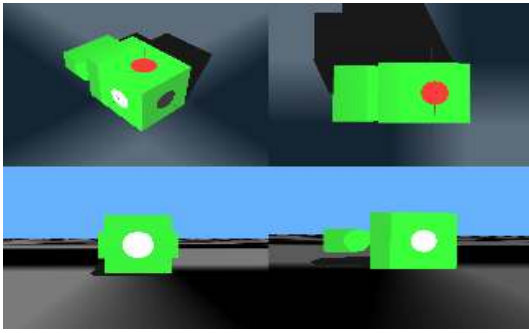


그림 1. Roombot의 모습. (a) 전체적인 모습, (b) 위에서 본 모습, (c) 정면에서 본 모습, (d) 측면에서 본 모습.

Fig. 1. Shape of Roombot. (a) full view, (b)bird eye's view, (c) front view, (d) side view.

모듈은 1개의 관절과 컨트롤러와 전원, 6개의 커넥터를 가지며, 1개의 DOF를 가진다. 모든 관절은 $-1.57 \sim 1.57$ 사이의 동일한 각도 제한을 가진다. 뱀형 로봇은 8개의 모듈로 구성되었으며, 앞 모듈의 후면 커넥터와 뒤 모듈의 전면 커넥터가 차례대로 결합되는 방식이다. 모든 모듈은 수직관절 형태를 가진다.

2. 모듈라 뱀형 로봇의 이동 방법

모듈라 뱀형 로봇의 이동 방법으로는 크게 수직적인 관절 움직임을 가지는 방법과 수평적인 관절 움직임을 가지는 방법, 수직과 수평 모두 움직일 수 있는 방법이 있다. 수평적인 관절의 움직임을 가지는 방법으로는 몸체의 측면 방향으로 전진하는 사이드와인더 방식[4,10], S모양으로 움직이는 측면 파동 방법(Serpentine)[3,10] 등이 있다. 수직적인 관절의 움직임을 가지는 방법은 일반적인 애벌레의 움직임과 같은 Caterpillar 방식[5,11], 꼬리부분으로 몸체를 밀어 이동하는 Inching 방식 등이 있다[10]. 그 외에 수평 수직 복합적인 움직임의 방법으로는 Helix Theory 방법[12]이나 Screw 방법[10,11] 등이 있다.

대부분의 모듈라형 뱀 로봇에서 수평관절은 수직관절의 보조로서 회전을 하는데 사용되고 있으며, 이동을 위한 경우는 모듈 밑에 바퀴가 있거나 수면 위를 이동하는 특수한 환경에서만 사용되고 있다. 따라서 본 연구에서도 수직관절의 조합에 의한 이동을 제어 대상으로 있다.

기존에 가장 널리 쓰이는 주기적 제어방식은 식 1과 같이 각 모듈마다 사인파에 일정한 위상차를 입력한다

$$\sin(t + n\phi) \quad (n = 0, 1, 2 \dots) \quad (1)$$

III. GA와 GP 기반 이동 제어

GA(Genetic Algorithm)[6]와 GP(Genetic Programming)[7]는 진화연산의 한 종류로서, 유전자(chromosome)를 표현하는 방식에 서로 차이가 있다. GA는 비트열이나 실수형태의 염색체를 이용하고, GP는 트리구조의 염색체를 이용한다. 이 차이로 인해 GA는 고정된 크기의 염색체로 구성되며, 주로

수치최적화에 적합하다. 이에 비해 GP는 트리구조를 사용하기 때문에 염색체의 크기가 가변이며, 구조적인 설계문제나 임의의 차수나 조합적으로 구성되는 복잡한 함수를 탐색하는 것에 적합하다. 본 논문에서는 GA 기반의 두가지 방식과 GP 기반의 방식등 총 3가지 방식을 비교한다.

1. GA 기반 단순 위상제어(GAPs)

이 방식은 식 (2)에 나타난바와 같이, 사인 함수의 위상(phase, ϕ)만을 가변 파라미터로 하여 움직임을 결정한다. 즉, 앞 절에 소개된 기존의 일반적인 제어방식은 고정된 위상값을 순차적으로 일정하게 증가시켜 사용하는 데 비해서, 이 방식은 모든 모듈이 최대와 최소 범위내에서 임의의 위상을 가질 수 있다. 개체는 8개의 관절에 대해서 각 1개의 파라미터를 가지고 총 8개의 염색체를 사용한다.

$$\sin(t + \phi) \quad (2)$$

$$(-6.28 \leq \phi \leq 6.28)$$

2. GA 기반 이동 제어

이 방법에서는 각 모듈에 대한 관절 값을 제어하기 위해서, 식 (3)과 같이 주어진 사인함수에 대해서, 진폭(A)과 주기(ω), 위상(ϕ)의 파라미터를 모두 결정한다.

$$A \sin(\omega(t) + \phi) \quad (3)$$

$$(-1.57 \leq A \leq 1.57, -1.5 \leq \omega \leq 1.5, -6.28 \leq \phi \leq 6.28)$$

한 관절(모듈)에 대해서 진폭(A), 주기(ω), 위상(ϕ)의 3개의 필드를 가지며, 개체는 8개의 관절로 이루어지므로 총 24개의 필드로 구성된 염색체를 사용한다.

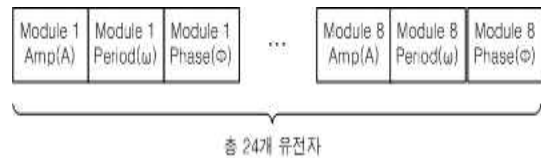


그림 2. 유전자(Chromosome)의 구성.

Fig. 2. Construction of Chromosome.

3. GP 기반 이동 제어

GA 기반 방법이 고정된 구조의 사인함수에 파라미터만을 변화시켜 이동 제어식을 탐색하는데 비하여, GP 방식에서는 각 모듈이 움직이는 관절 궤적이 임의의 함수로 구성된다. 즉, 그림 3과 같이 각 관절에 대한 궤적 함수가 GP에 의해서 생성, 진화되는 트리를 통해 탐색된다[13].

각 관절의 독립적인 제어를 위해서, 일반적인 GP의 단일 트리 방식이 아닌 멀티트리 방식을 이용함으로써 더 효율적인 연산을 꾀하였다[13].

GP 트리의 구성에 이용된 함수와 터미널은 다음과 같다.

$$F = \{ \sin, \cos, +, -, *, / \}$$

$$T = \{ X, ERC \}$$

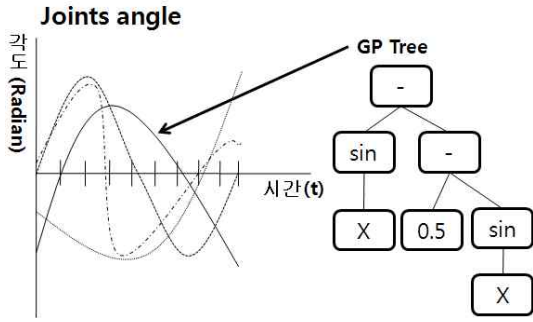


그림 3. GP Tree를 통한 움직임 생성.
Fig. 3. Design of Locomotion with GP tree.

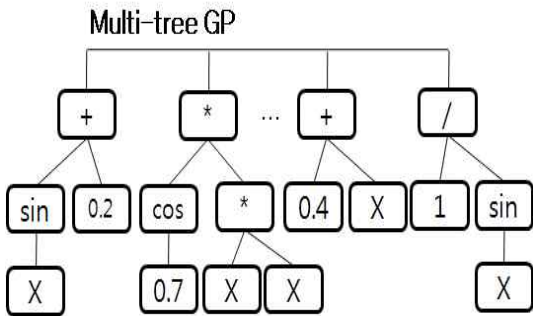


그림 4. 멀티트리 GP 방식.
Fig. 4. Multi-tree GP.

IV. 컴퓨터 시뮬레이션

1. Webots 시뮬레이션 환경

시뮬레이션 환경은 Cyberbotics사의 Webots[8]을 사용하였다. Webots은 모바일 로봇에 대한 모델링, 프로그래밍, 그리고 시뮬레이션 기능을 제공하는 모바일 로봇 시뮬레이션 S/W 이다. 주요 기능은 다음과 같다. 각종 센서와 액츄에이터에 대한 라이브러리 제공, 정확한 Physics 시뮬레이션을 위한 ODE(Open Dynamics Engine) 라이브러리 제공, 그리고 실제 모바일 로봇에 컨트롤러를 TRANSPER 할 수 있는 기능을 가지고 있다.

2. 적합도 함수 및 GA, GP 파라미터

이동 제어의 성능을 측정하는 척도는 이동 거리를 기본으로 하였고, 직진성을 우선하기 위해서 벗어난 거리(z)를 벌칙으로 두었다. 실험에 사용된 적합도 함수는 다음 식 (4)와 같이 구성하였다.

$$fitness = (0.9 \times x - 0.4 \times |z|) \tag{4}$$

GA 프로그램으로는 SGA[7]가 사용되었고, GP 엔진으로 lil-gp[14]가 사용하였다. GA와 GP에 공통으로 이용된 파라미터는 다음과 같다.

- Number of generations : 50
- Population sizes : 20, 30
- Crossover : 0.9
- Mutation : 0.1
- Selection : Roulette Wheel

GP의 초기 트리 생성에 대한 파라미터는 다음과 같다.

- Initial population : half_and_half
- Initial depth : 1-3
- Max depth : 12

3. 실험 결과 및 분석

각각의 생성된 함수에 대해 GA, GP 모두 한 사이클당 40개의 보간점을 이용하였다. 각각 10회씩 반복 실험하였으며, 적합도가 클수록 단위시간당 이동 거리가 긴 것을 의미한다.

실험은 뱀 로봇의 수직 이동 방식을 이용하였고, 성능지표는 정의된 적합도와 속도(cm/s)로 측정하였다. 시뮬레이션은 10회 반복 수행하여 평균값을 기록하였다.

먼저, 평지에서 실험한 기존의 주기적 위상(Phase) 접근법에 의한 결과는 다음과 같다. 식 (1)에서 ϕ 를 1.2로 사용한 경우가 가장 속도면에서 우수했으며, 시뮬레이션 결과는 적합도 1.3, 속도는 20.31 cm/s로 느리게 나타났다. 그림 5에 이동 형태가 나와 있으며, 각 모듈 관절각의 변화가 적은 매우 기본적인 이동 모습을 보이고 있다.

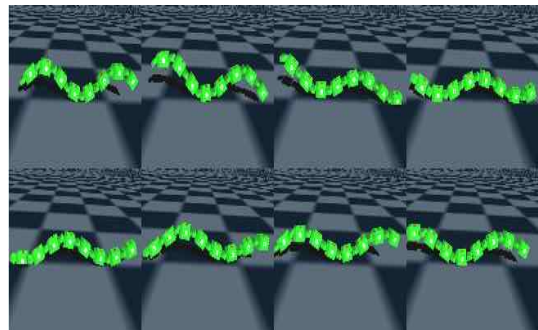


그림 5. 위상 방식의 이동 움직임.
Fig. 5. A phase locomotion of snakebot.

3.1 평지환경에서의 실험

III 절에서 제안된 3가지 방법에 대한 평지환경 실험 결과가 표 1-4에 나와 있다. 먼저 표 1과 2에는 개체수 20에 대한 적합도와 속도 결과가 나와 있고, 표 3과 4에는 개체수 30에 대한 성능 결과가 나와 있다.

표 1. 20 개체 실험에 대한 적합도 수치.

Table 1. Fitness values of experiments for 20 population size.

	Phase	GAps	GP	GA
Avg	1.3	1.88	2.54	2.76
표준편차	0.0	0.50	0.41	0.84

표 2. 20 개체 실험에 대한 속도.

Table 3. Speed values of experiments for 20 population size.

	Phase	GAps	GP	GA
Avg	20.31	29.41	39.64	43.18
표준편차	0.0	7.89	6.45	13.16

표 3. 30 개체 실험에 대한 적합도 수치.

Table2. Fitness values of experiments for 30 population size.

	Phase	GAps	GP	GA
Avg	1.3	2.17	2.72	3.06
표준편차	0.0	0.38	0.49	0.84

표 4. 30 개체 실험에 대한 속도.

Table4. Speed values of experiments for 30 population size.

	Phase	GAps	GP	GA
Avg	20.31	34.01	42.59	47.75
표준편차	0.0	6.04	7.69	13.16

표 1-4에 나타난 결과 값을 종합해 볼 때 속도와 적합도 모두 GA > GP > GAps(phase만 제어) 순으로 나타나서, GA에 의한 결과가 가장 우수함을 알 수 있다. 또한, 모든 방법에 대해서 개체수가 30인 경우가 개체수가 20인 경우에 비해서 더 높은 적합도와 속도를 나타냄을 알 수 있다.

그림 6-9에는 위상 파라미터만을 GA로 탐색한 GAps 방법에 대한 이동 움직임 화면과 각 관절각들의 그래프가 나와 있다.

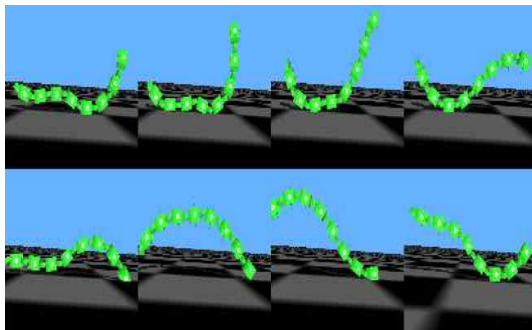


그림 6. 최우수 개체의 이동 움직임. (GAps, popsize=20)
Fig. 6. Locomotion of best individual. (GAps, popsize=20)

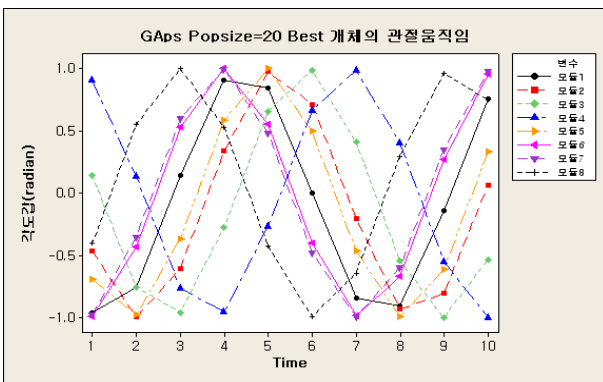


그림 7. 관절 궤적 그래프. (GAps, popsize=20)
Fig. 7. Joints trajectory graph (GAps, popsize=20)

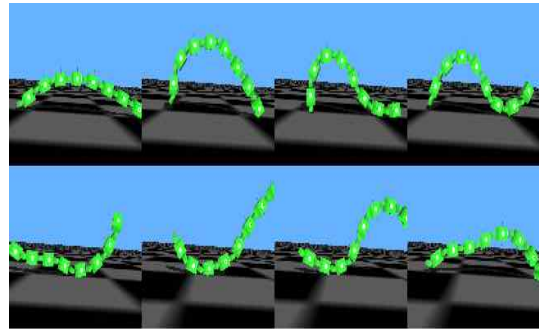


그림 8. 최우수 개체의 이동 움직임. (GAps, popsize=30)
Fig. 8. Locomotion of best individual. (GAps, popsize=30)

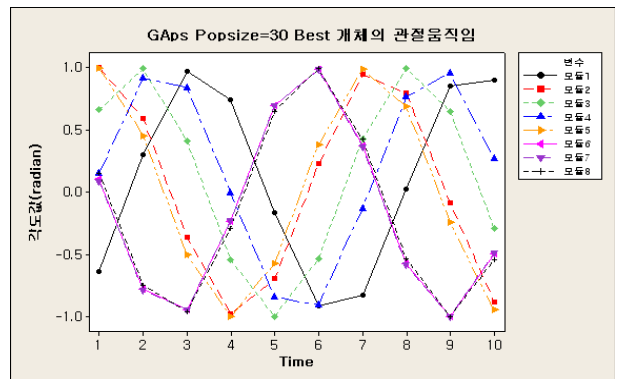


그림 9. 관절 궤적 그래프. (GAps, popsize=30)
Fig. 9. Joints trajectory graph (GAps, popsize=30)

그림 10-13에는 GP를 사용한 방법과 그림 14-17에 GA 기법에 대한 이동 움직임 화면과 각 관절각들의 그래프가 나와 있다.

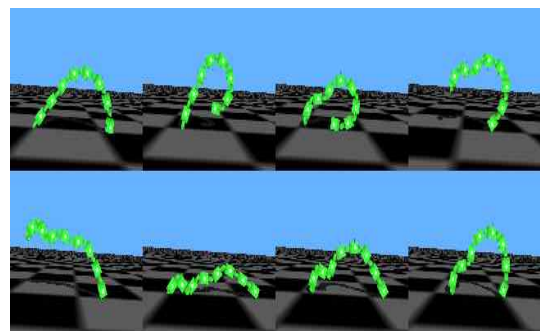


그림 10. 최우수 개체의 이동 움직임. (GP, popsize=20)
Fig. 10. Locomotion of best individual. (GP, popsize=20)

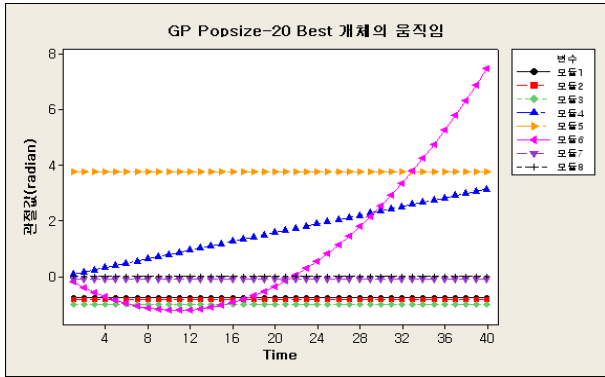


그림 11. 관절 궤적 그래프. (GP, popsize=20)
 Fig. 11. Joints trajectory graph. (GP, popsize=20)

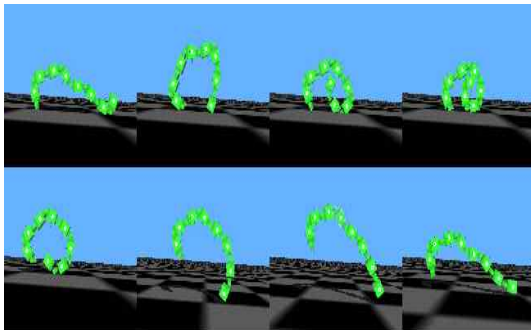


그림 12. 최우수 개체의 이동 움직임. (GP, popsize=30)
 Fig. 12. Locomotion of best individual. (GP, popsize=30)

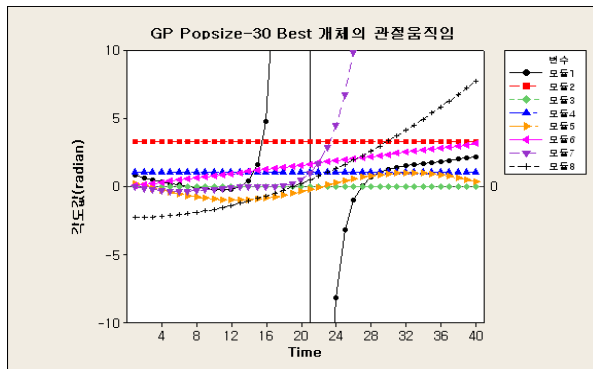


그림 13. 관절 궤적 그래프. (GP, popsize=30)
 Fig. 13. Joints trajectory graph. (GP, popsize=30)

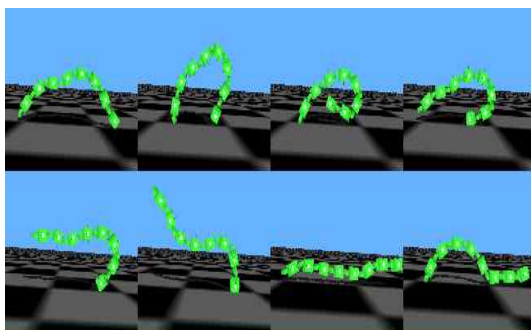


그림 14. 최우수 개체의 이동 움직임. (GA, popsize=20)
 Fig. 14. Locomotion of best individual. (GA, popsize=20)

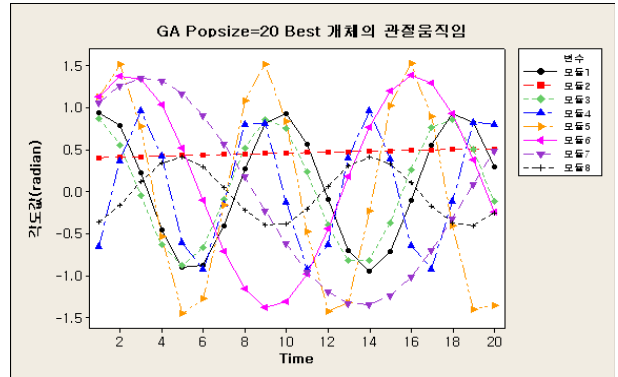


그림 15. 관절 궤적 그래프. (GA, popsize=20)
 Fig. 15. Joints trajectory graph. (GA, popsize=20)

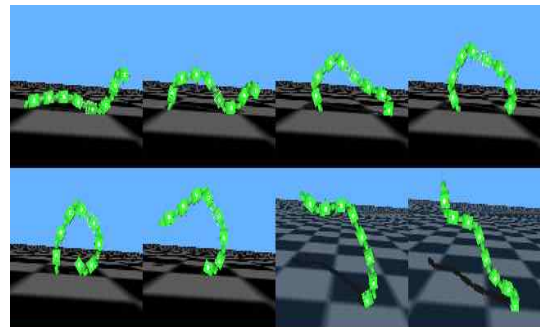


그림 16. 최우수 개체의 이동 움직임. (GA, popsize=30)
 Fig. 16. Locomotion of best individual. (GA, popsize=30)

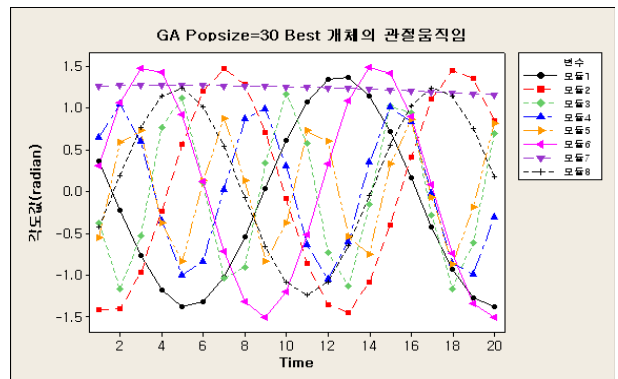


그림 17. 관절 궤적 그래프. (GA, popsize=30)
 Fig. 17. Joints trajectory graph. (GA, popsize=30)

이동 움직임과 관절궤적을 분석해보면, GAs의 경우는 모든 관절이 골고루 사용되었고, 이동 모양도 모두 사인곡선을 그대로 따라가는 모습을 보이며 가장 안정한 움직임을 나타내고 있다.

GP의 경우에는 관절간의 편차가 많이 나타나며, 어떤 관절은 거의 사용되지 않는 모습을 보인다(그림 11, 13). 이동 형태를 보면 주로 탄성을 이용한 걸음새로서, 모듈라 로봇이 둥글게 말린 모습을 했다가 마지막 모듈을 작용점으로 사용하여 펼쳐나가는 움직임을 보인다.

GA에서의 관절 궤적은 거의 GAs와 유사하지만, 역시

한 두개 모듈이 일정한 값을 가지며, 각 모듈간의 진폭에도 GAs보다는 차이가 있음을 나타낸다(그림 14, 16). 또한 움직임의 형태에 있어서는 GP와 유사하게 둥글게 말았다가 펼치는 형식을 취하고 있지만, 후반부에는 또 다른 도약을 하는 움직임을 보인다.

GAs와 GA는 사인함수를 기본으로 하여 관절케지의 패턴을 변형한 것이므로, GP에 비해서 안정적인 움직임을 나타낸 것으로 생각된다. 그리고, GP와 GA에 의한 결과는 좀 더 큰 움직임을 통해 이동 거리를 늘릴 수 있는 관절케지의 임의조합을 탐색하기 때문에 적합도와 속도가 우수한 결과를 얻은 것으로 파악된다. 또한 일반적으로 탐색시 GP는 GA보다 많은 개체수를 필요로 하는 경향이 있고, 가변적인 트리의 특성상 관절식에 대한 제한이 거의 없으므로 GA에 비해 탐색 공간이 매우 넓다. 그 결과 현재까지의 제한된 실험에서는 GA보다 약간 낮은 성능을 보이는 것으로 판단된다.

3.2 경사 및 계단 환경에서의 실험

평지 이동외에 난이도가 높은 경사형과 계단형의 두 가지 실험을 추가하였다. 진화연산의 다른 파라미터는 모두 동일한 상태에서 개체수는 30을 사용하였다. 경사형의 경우는 15도의 경사를 가진 환경을 구성했고, 계단형의 경우 높이는 모듈의 높이와 동일하고 너비는 모듈 높이의 3배인 계단을 구성하였다.

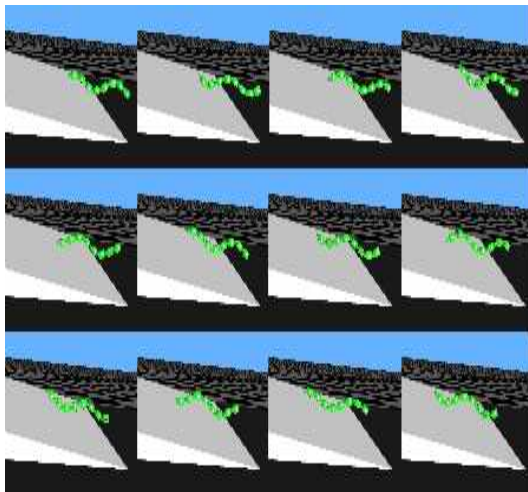


그림 18. 경사환경에서의 위상 방식의 이동 움직임
Fig. 18. A phase locomotion of snakebot (Slope environment).

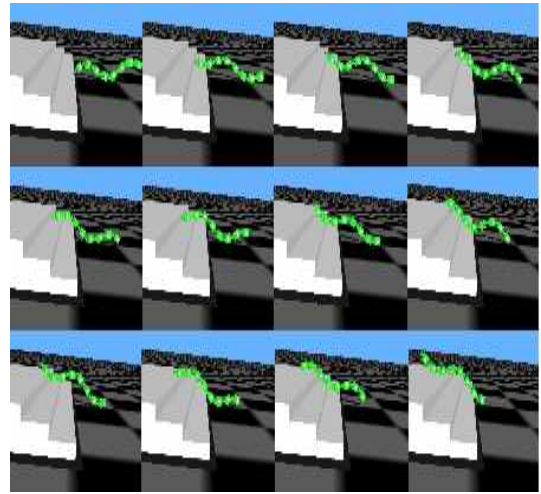


그림 19 계단환경에서의 일반적인 방식의 이동 움직임
Fig. 19. A phase locomotion of snakebot (Slope environment).

표 5. 경사환경 실험에 대한 적합도 수치.

Table 7. Fitness values of experiments for Slope environment..

	Phase	GAs	GP	GA
Avg	1.87	2.83	2.14	2.66
표준편차	0.0	0.37	0.07	0.25

표 6. 경사환경 실험에 대한 속도.

Table 8. Speed values of experiments for Slope environment..

	Phase	GAs	GP	GA
Avg	21.4	44.25	33.41	41.61
표준편차	0.0	5.70	1.15	3.89

표 7. 계단환경 실험에 대한 적합도 수치.

Table 5. Fitness values of experiments for Stairs environment..

	Phase	GAs	GP	GA
Avg	0.8	2.36	1.44	2.14
표준편차	0.0	0.65	0.06	0.69

표 8. 계단환경 실험에 대한 속도.

Table 5. Speed values of experiments for Stairs environment..

	Phase	GAs	GP	GA
Avg	12.5	24.95	22.52	23.59
표준편차	0.0	12.48	1.01	11.76

위의 두 환경에 대한 실험 결과는 표 5-8에 나타나있다. 전체적으로, 평지에서의 결과와 비교했을 때 어려운 환경에서의 이동이 더 저하됨을 알 수 있다. 그리고, 평지실험 결과와는 다르게 속도 및 적합도가 GAs > GA > GP 순으로 나타났다. 즉, 어려운 환경에서는 GAs와 같이 반복성이 높은 이동 움직임이 더 뛰어난 결과를 보임을 알 수 있다. 또한 계단환경과 경사환경중에서는 계단환경이 더 난이도가 높음을 보여준다.

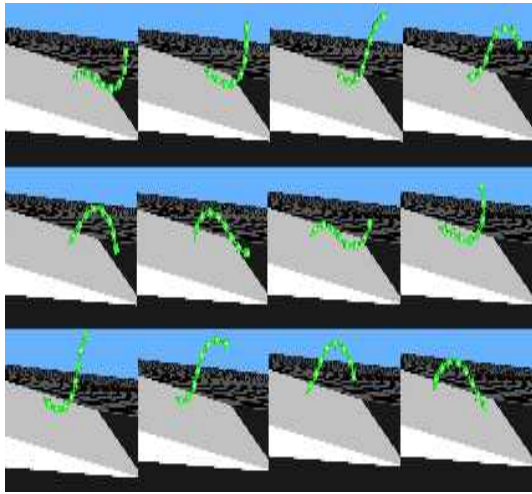


그림 20. 최우수 개체의 이동 움직임. (Gaps, 경사환경)
 Fig. 20. Locomotion of best individual. (Gaps, Slope environment)

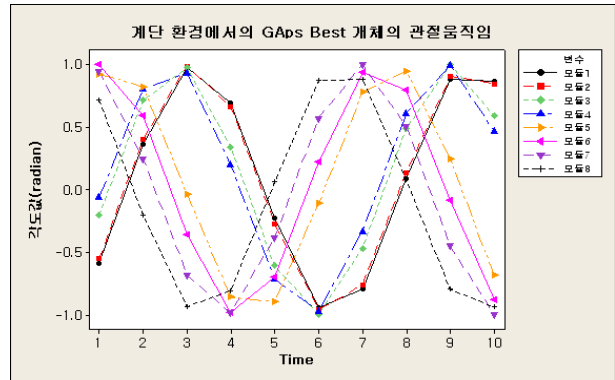


그림 23. 관절 궤적 그래프. (Gaps, 계단환경)
 Fig. 23. Joints trajectory graph. (Gaps, Stairs environment)

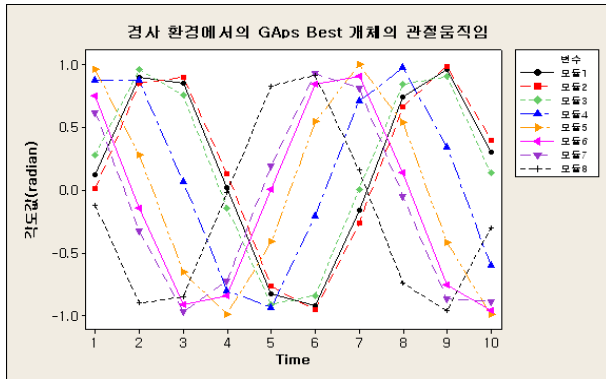


그림 21. 관절 궤적 그래프. (Gaps, 경사환경)
 Fig. 21. Joints trajectory graph. (GA, Slope environment)

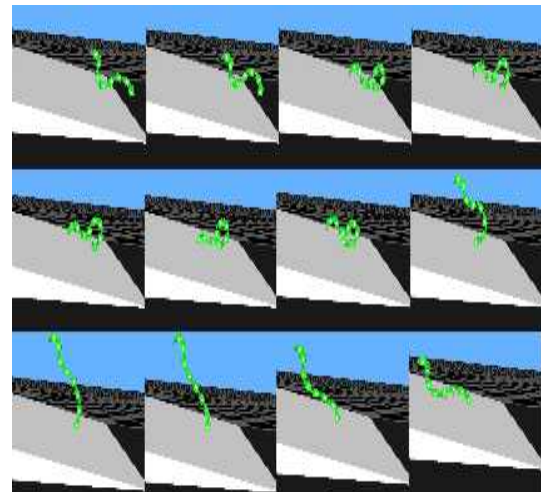


그림 24. 최우수 개체의 이동 움직임. (GP, 경사환경)
 Fig. 24. Locomotion of best individual. (GP, Slope environment)

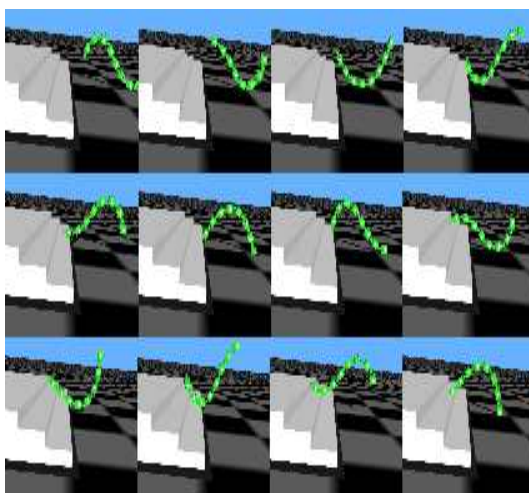


그림 22. 최우수 개체의 이동 움직임. (Gaps, 계단환경)
 Fig. 22. Locomotion of best individual. (Gaps, Stairs environment)

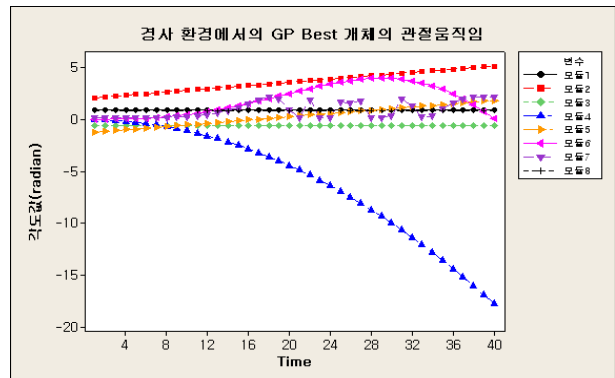


그림 25. 관절 궤적 그래프. (GP, 경사환경)
 Fig. 25. Joints trajectory graph. (GP, Slope environment)

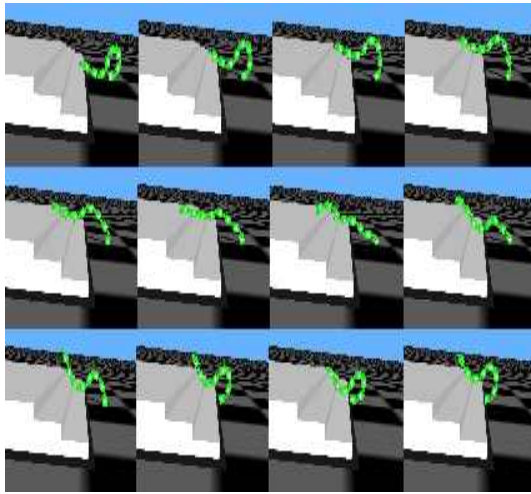


그림 26. 최우수 개체의 이동 움직임. (GP, 계단환경)
 Fig. 26. Locomotion of best individual. (GP, Stairs environment)

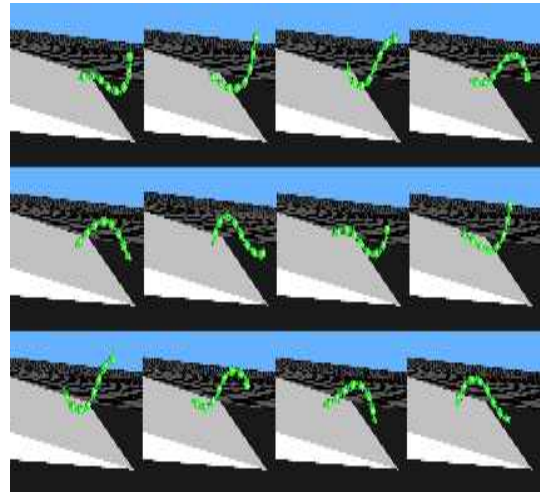


그림 28. 최우수 개체의 이동 움직임. (GA, 경사환경)
 Fig. 28. Locomotion of best individual. (GA, Slope environment)

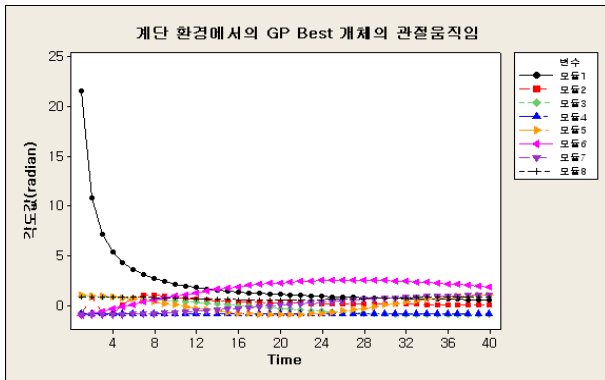


그림 27. 관절 궤적 그래프. (GP, 계단환경)
 Fig. 27. Joints trajectory graph. (GP, Stairs environment)

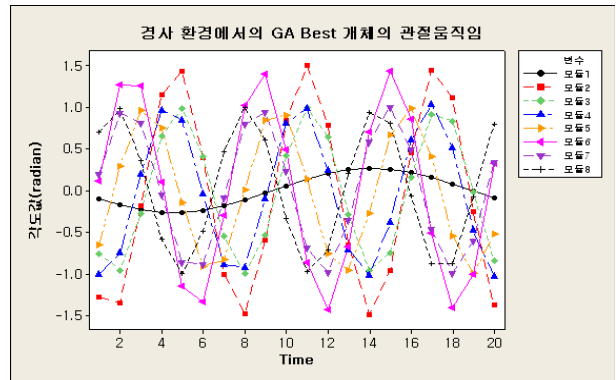


그림 29. 관절 궤적 그래프. (GA, 경사환경)
 Fig. 29. Joints trajectory graph. (GA, Slope environment)

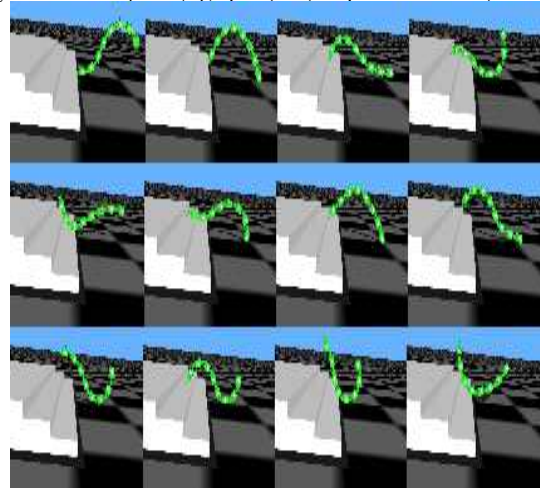


그림 30. 최우수 개체의 이동 움직임. (GA, 계단환경)
 Fig. 30. Locomotion of best individual. (GA, Stairs environment)

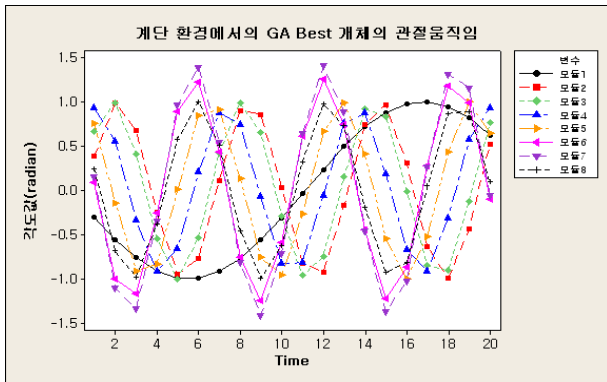


그림 31. 관절 궤적 그래프. (GA, 계단환경)
 Fig. 31. Joints trajectory graph. (GA, Stairs environment)

그림 20-31에는 3가지 방법에 의해 얻어진 이동 움직임 모습과 각 관절자들의 그래프가 나와 있다. GAs의 경우, 이동 형태가 사인곡선에 제일 가까운 모습을 보인다(그림 20, 22). 그리고, 각 관절마다 거의 일정한 위상 차이를 가짐을 알수있다(그림 21, 23), GP의 경우 이동 모습 및 관절각 그래프가 평지에서와 유사하거나 좀 큰 움직임을 나타냈다(그림 24-27). GA에서는 GAs와 비슷한 사인파에 가까운 모양새를 나타냈다. 평지에서 보였던 탄성을 이용한 움직임과는 차이가 큰 움직임을 보였다(그림 28, 30). 관절 그래프에서도 평지에서와 달리 일정한 값을 가지는 모듈이 없었고, 1번 모듈의 사용이 적고 6번 모듈의 사용이 많은 것을 보여준다. 또한 거의 일정한 위상을 가진 GAs와는 다르게 관절각마다 위상 차이가 나타났었다(그림 29, 31).

평지에서는 GA에 의한 결과가 가장 우수한테 비해, 계단과 경사환경에서는 GAs가 제일 뛰어난 결과를 보였다. 이는 계단을 넘거나, 경사를 올라가는 환경에서는, 모든 관절이 일정한 값을 가지고 위상차이가 일정한 경우가 더 적합함을 보여준다.

VI. 결론

본 논문에서는 모듈라 뱀형 로봇에 GA와 GP를 이용한 이동 제어를 통해 3가지 방법에 대한 성능 결과와 차이점에 대해서 분석하였다. 평지에서는 평균적으로 주요 성능지표인 속도 면에서 GA 방식이 가장 우수함을 나타내고, 이에 비해 GP의 경우는 움직임 면에서는 비교적 동작이 크고 정형화에서 벗어난 결과를 보인다. 마지막으로 GAs는 속도는 가장 느리지만, 안정적인 움직임을 나타내었다. 반면에 어려운 환경(경사, 계단)에서는 GAs 방식이 가장 우수했고, GA가 그 다음으로 우수했다. GP가 속도, 적합도 면에서 가장 낮은 결과를 가지지만, 다른 방식처럼 일정하고 비슷한 움직임이 아닌 독특한 움직임을 보였다. 이는 평지와 같이 안정된 환경에서는 각 관절의 이동성이 더욱 클수록 이동이 용이하고, 어려운 환경에서는 각 관절이 안정적이고 반복적인 움직임을 가질수록 이동이 용이하다는 것을 의미한다.

기존의 진화기법을 응용한 기법들은 대부분 사인파 형태의 안정된 움직임을 중심으로 변형된 이동 형태인데 비해,

본 연구에서는 각 모듈관절의 탐색범위를 제한치 않고 이동 제어를 한 결과, 뛰어난 속도 및 특이한 형태의 이동 형태도 얻을 수 있었다. 또한 여러 방식을 사용한 실험의 비교를 통하여 진화연산의 특정 방식이 모든 환경에 다 우수하기는 어려우며, 환경에 적합한 진화 방식이 존재함을 발견하였다.

향후, 이를 기반으로 다양한 적합도 함수에 대해서 좀 더 정교한 탐색기법 연구와 모듈라 로봇의 특징인 일부 모듈 손상시의 이동 제어 및 실제 로봇에 대한 적용이 필요하다고 사료된다.

참고문헌

- [1] G. S. Hornby, H. Lipson, J. B. Pollack, "Generative Representations for the Automated Design of Modular Physical Robots", *IEEE Trans. on Robotics and Automation*, vol. 19, no. 4, pp. 703-719, 2003.
- [2] V. Zykov, E. Mytilinaios, M. Desnoyer, H. Lipson, "Evolved and Designed Self-Reproducing Modular Robotics", *IEEE Trans. on Robotics*, vol. 23, no. 2, pp. 308-319, 2007.
- [3] A. Crespi, A. J. Ijspeert, "AmphiBot II : An Amphious Snake Robot that Crawls and Swims using a Central Pattern Generator", *In Proceedings of the 9th International Conference on Climbing and Walking Robots (CLAWAR 2006)*, pp. 19-27, Brussels, Belgium, September 2006.
- [4] I. Tanev, "Genetic Programming Incorporation Biased Mutation for Evolution and Adaptation of Snakebot", *Genetic Programming and Evolvable Machines*, vol. 8, no. 1, pp. 39-59, 2007.
- [5] Mark W. Sherman, "Sine-Wave Locomotion in a Robotic Snake Model Form and Programming", *In Proceedings of AAAI Mobile Robot Competition: Papers from the AAAI Workshop*, pp. 34-40, American Association for Artificial Intelligence, Menlo Park, California, 2002.
- [6] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, MA, 1989.
- [7] J. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT press, 1992.
- [8] L. Hohl, R. Tellez, O. Michel, A. J. Ijspeert, "Aibo and Webots: Simulation, wireless remote control and controller transfer", *Robotics and Autonomous systems*, vol. 54, pp. 472-485, 2006.
- [9] A. Billard, P. Dillenbourg, Roombots, <http://birg.epfl.ch/page65721.html>.
- [10] K. J. Dowling, *Limless Locomotion: Learning to Crawl with a Snake Robot*, in his Ph.D Thesis, Robotics Institute, Carnegie Mellon University 1997.
- [11] H. Choset, Modular Snake Robots,

<http://download.srv.cs.cmu.edu/~biorobotics/projects/modsnake/modsnake.html>.

- [12] I. Tanev, T. Ray, K. Shimohara, “On the Analogy in the Emergent Properties of Evolved Locomotion Gaits of Simulated Snakebot”, *Mobile Robots Toward New Applications*, ch. 19, pp. 559-578, Germany, December 2006.
- [13] 서기성, 현수환, “관절 공간에서의 GP 기반 진화기법을 이용한 4족 보행로봇의 걸음새 자동생성”, *제어 · 로봇 · 시스템학회 논문지*, 제 14권 6호, pp. 573-579, 2008. 6.
- [14] D. Zongker, B. Punch, *Lil-GP User's Manual*, Michigan State University, July 1995.