

Evolutionary Optimization of Neurocontroller for Physically Simulated Compliant-Wing Ornithopter

Yoonsik Shim*

*Research Professor, Institute of Computer, Information and Communication, Korea University, Seoul, Korea

[Abstract]

This paper presents a novel evolutionary framework for optimizing a bio-inspired fully dynamic neurocontroller for the maneuverable flapping flight of a simulated bird-sized ornithopter robot which takes advantage of the morphological computation and mechansensory feedback to improve flight stability. In order to cope with the difficulty of generating robust flapping flight and its maneuver, the wing of robot is modelled as a series of sub-plates joined by passive torsional springs, which implements the simplified version of feathers attached to the forearm skeleton. The neural controller is designed to have a bilaterally symmetric structure which consists of two fully connected neural network modules receiving mirrored sensory inputs from a series of flight navigation sensors as well as feather mechanosensors to let them participate in pattern generation. The synergy of wing compliance and its sensory reflexes gives a possibility that the robot can feel and exploit aerodynamic forces on its wings to potentially contribute to the agility and stability during flight. The evolved robot exhibited target-following flight maneuver using asymmetric wing movements as well as its tail, showing robustness to external aerodynamic disturbances.

▶ **Key words:** Flapping flight, Evolutionary robotics, Neural network, Simulation, Compliance

[요 약]

본 논문은 목표한 방향으로 자유롭게 기동할 수 있는 새 크기의 물리기반 날갯짓 비행로봇 시뮬레이션을 위한 동역학적 신경망 컨트롤러를 생성하는 통합적인 진화연산 방법을 제시한다. 제안된 진화로봇 시스템은 날갯짓 비행의 추가적인 민첩성과 안정성을 위하여 Morphological Computation 개념을 응용한 간단한 날개 순응성 모델과 그와 통합된 Mechanosensory 정보를 활용한다. 역학적으로 불안정한 날갯짓 기동의 안정성 개선을 위해 로봇의 날개는 회전스프링으로 팔의 골격에 연결된 여러개의 패널들로 모델링되어, 새의 깃털에서 영감을 받은 단순한 형태의 날개 유연성을 시뮬레이션 하도록 설계되었다. 신경망 컨트롤러 역시 생물학적으로 의미있는 좌우대칭적 연결구조를 가짐과 동시에 최대의 진화연산 탐색 가능성을 위해 두 개의 fully-connected 신경망 모듈로 이루어지며, 이를 위한 센서정보로서 항법센서와 더불어 각 날개패널의 움직임 정보들이 입력되어진다. 이러한 설계는 각 패널센서로 하여금 잠재적으로 신경망의 날갯짓 패턴 생성에 관여하게 함과 동시에, 날개에 가해지는 힘의 감지와 패널의 굽어짐으로 인한 날개 순응성으로부터 얻을 수 있는 비행의 민첩성과 안정성 향상을 동시에 유도할 수 있다. 본 시스템으로 진화된 날갯짓 로봇은 실시간으로 주어지는 목표방향으로의 효과적인 기동과 함께, 외부의 공기역학적 섭동에 대하여도 더욱 안정적인 비행을 유지함을 보여준다.

▶ **주제어:** 날갯짓 비행, 진화로보틱스, 신경망, 시뮬레이션, 순응성

-
- First Author: Yoonsik Shim, Corresponding Author: Yoonsik Shim
 - *Yoonsik Shim (kdhong@south.ac.kr), Institute of Computer, Information and Communication, Korea University
 - Received: 2019. 12. 02, Revised: 2019. 12. 26, Accepted: 2019. 12. 26.

I. Introduction

지구상에서는 수백만 년 동안 곤충으로부터 익룡에 이르기까지 다양한 비행 생물들이 진화하였으며, 그중에서도 조류는 지금까지 살아남은 가장 진보된 비행종 중 하나라고 할 수 있다. 빠르고 고정된 날갯짓 각도를 보이는 곤충들과 달리 새들은 상대적으로 느린 날개움직임을 가지며, 한 번의 날갯짓 동안 연속적으로 관절의 각도와 날개의 모양을 조절해야만 공중에서의 균형을 유지할 수 있다[1]. 소형 드론의 필요성의 대두로 인해 활발한 연구가 진행되고 있는 곤충 크기의 날갯짓 비행에 비하여, 조류 이상의 크기를 가지는 날갯짓 비행체의 움직임을 생성하기 위한 연구는 상대적으로 부족한 현실이다. 로보틱스 및 공학 분야에서 현재까지의 대부분의 날갯짓 비행로봇 컨트롤의 구축은 전방으로 순항시의 공기역학의 분석에 중점을 두고 있으며, 방향전환 및 예기치 못한 외력에 대한 대응과 관련해서는 pitch 모멘트 안정성 또는 단순히 꼬리날개의 독립적인 조절을 통한 방향전환 등 사전 지식에 의존한 인위적이고 직관적인 해법들이 주로 연구되어왔다[2,3]. 그러나 날갯짓 비행에서의 가장 큰 도전은 민첩하면서도 안정적인 기동이며 공기역학 환경이라는 가장 민감한 안정성을 가진 상황에서 예기치 못한 상황에 대응하는 알맞은 컨트롤러 시스템을 자동으로 생성하는 것은 아직도 풀어야 할 숙제이다.

인위적인 디자인을 최소화하고, 자동으로 날갯짓 컨트롤러를 최적화하기 위한 시뮬레이션 연구들은 대부분 단순한 곡선 함수만을 이용하거나 이를 보완하기 위한 Feedforward Neural Network (FFNN) 모델을 함께 사용하는 방식으로 성공적인 날갯짓 기동을 생성하여 왔다[4-6]. 특히 최근 계산능력의 향상에 따라 컴퓨터 애니메이션 분야를 중심으로 Deep Reinforcement Learning (DRL)을 활용하여 모델독립적 학습방법을 통한 날갯짓 비행의 생성이 연구되었으며 모션캡처 데이터 또는 사용자가 제공한 움직임 정보를 기반으로 학습하거나, 사용자가 정의한 경로를 따라 움직이도록 학습하는 방법으로 물리 기반 애니메이션을 위한 성공적인 컨트롤러를 생성하였다[7,8]. 이러한 딥러닝과 강화학습 기반 학습방법이 많은 사지기동 컨트롤 생성 분야에서도 좋은 성능을 보여주고 있으나, 대부분 큰 규모의 FFNN을 이용하여 매우 많은 수의 state 및 sensor 정보와 함께 수 많은 경험 데이터를 수집하고, 학습에 관련된 hyperparameter들의 세심한 조절 및 local optima와 credit assignment 문제들의 해결 등 복잡한 과정들을 거쳐야 하는 어려움이 있다.

로봇 컨트롤 생성분야의 또 다른 줄기로서 gradient free 확률적 파라미터 최적화 방법이 오랫동안 사용되어 왔는데, 그중 하나로 진화연산을 사용하는 진화로보틱스 (Evolutionary Robotics)는 대상 모델에 대한 제약이 없는 관계로 생물학에서 영감을 받은 다양한 종류의 높은 비선형적 컨트롤 모델들을 적용할 수 있다. 이 중 사용자의 사전지식을 최소화하면서도 모든 복잡한 비선형적 패턴을 만들어낼 수 있는 Recurrent Neural Network (RNN) 모델이 주로 쓰이고 있는데, 진화연산을 사용하여 이러한 신경망의 연결강도의 조절을 넘어서 그 구조까지도 동시에 최적화시킬 수 있는 Open-ended 시스템의 생성에도 많은 가능성을 보여왔다. 또한 최근의 연구들에서, 간단한 진화알고리즘으로도 ATARI 게임 등 state-of-art DRL 시스템과 대등한 성능을 보일 수 있음이 밝혀졌고[9], 현재로서는 동일한 문제에 대한 진화연산의 curse of dimensionality 문제가 지적되고 있으나, 진화연산 특유의 높은 병렬화 용이성은 미래의 분산처리 성능의 발전에 따라 기존의 DRL보다 훨씬 높은 계산효율 증가도를 기대할 수 있다는 것이 입증되어 있다[10].

Recurrent neural network을 활용한 로봇 컨트롤 생성 방법은 최근의 DRL을 중심으로 한 기계학습에 쓰이는 FFNN 모델과 달리 높은 비선형성과 함께 자연의 신경계에 존재하는 사지기동 (limed locomotion)에 필수적인 자동적인 oscillatory 패턴 생성을 위한 Central Pattern Generator (CPG)의 구현이 가능하다. 이러한 비선형 동역학 시스템에서 나타나는 갈래질 (bifurcation) 현상을 통하여 같은 컨트롤러에 가해지는 센서입력 또는 외부제어신호에 따라 완전히 질적으로 다른 신호패턴들을 만들어낼 수 있는 이점이 있다. 이는 생물학적으로도 타당할 뿐 아니라 로보틱스 분야에 있어서 양적으로 거대한 Deep FFNN을 사용할 필요없이, 작은규모의 신경망 구조로도 매우 복잡한 행동패턴을 만들어 낼 수 있는 모델이라고 할 수 있다. 또한 본 연구에서와 같은 비행로봇의 특성상 가벼운 몸체를 위해 매우 제한된 계산능력의 컨트롤 모듈을 탑재할 수 밖에 없는 경우를 생각할 때, 컨트롤 모델 규모의 최소화가 용이하다는 점도 진화연산+RNN 방법의 장점이라고 볼 수 있다.

진화로보틱스를 통한 3차원 공간 상에서 자유롭고 기동성 있는 날갯짓 비행체의 신경망 컨트롤 생성에 대한 연구는 그리 많지 않은데, Mouret[11,12]은 단계적인 목표행동들을 순차적으로 진화시키는 방식으로 물리기반 새 로봇을 위한 신경망을 생성하였다. 첫단계로 고도를 유지하는 직진순항을 위해 대칭적 날개움직임을 위한 RNN 신

경망구조를 생성시킨 뒤 다음단계로 꼬리날개의 조절을 사용한 방향전환 컨트롤을 학습하였다. 신경망 컨트롤러 뿐 아니라 새 로봇의 날개와 몸통 등 각 신체부위의 크기와 모양을 동시에 진화시키는 방법도 연구되었는데[5,6] 이는 비행체의 특성상 Aspect Ratio, Wing Loading 등 날개모양과 비행체 무게등의 물리적 관계가 비행효율 날개움직임 패턴에 큰 영향을 미치기 때문이다. 이들은 방향전환을 위해 주로 꼬리날개의 조절에 중점을 두었으나, 실제 새의 날갯짓 기동은 양 날개의 비대칭적인 움직임으로 인한 모멘트의 생성이 주 역할을 하고 있으며, 이착륙을 제외한 대부분의 비행속도에서 꼬리날개는 접힌채로 유지되어 그 역할이 미미하다는 것이 알려져 있다[13].

본 연구는 이러한 단계적인 최적화 없이도, 한번의 연산을 사용하여 목표 지점에서의 방향전환 및 고도조절이 가능한 신경망 컨트롤러를 진화시키는 진화학습 프레임에 제안한다. 이를 위해 본 연구는 생물학적으로 의미있는 간단한 개념의 대칭적 RNN 구조를 제안하여 최적화해야 할 신경망 파라미터 수를 반으로 줄이면서도 효과적인 날갯짓 기동을 보일 수 있도록 하였다. 이에 더하여 동역학적으로 불안정한 날갯짓 로봇의 상충하는 문제들인 기동성과 안정성의 성공적인 조율을 위해 체화지능 (Embodied Intelligence) 분야의 방법론인 형태계산 (Morphological Computation) [14]을 구현하는 Wing compliance 및 Mechanosensory 정보를 도입하였다. 이를 통한 날개의 외력에 대한 순응성은 민감한 공기역학 환경 하에서 안정성을 높일 수 있고, 그에 통합된 날개의 휘어짐 센서들이 동시에 날갯짓 패턴 생성에 적극적으로 관여하도록 유도함으로써 애기치 못한 외부의 동기역학적 섭동이 주어졌을 때 이를 감지할 수 있는 컨트롤 구조를 만들 수 있다. 이를 구현하기 위해 여러개의 passive torsional spring으로 연결된 패널들로 이루어진 날개모델을 도입하여, 외력에 대해 굽어질 수 있는 유연한 물리적 특성과 함께 굽어짐의 정도를 측정할 수 있도록 하는 간단하면서도 분석과 구현이 용이한 모델을 제시한다. 이후 섹션에서 각 부분의 자세한 기술과 함께 결과 및 논의를 기술하려 한다.

II. Simulated Ornithopter Model

날갯짓 로봇은 몸통에 연결된 두 개의 팔에 여러개의 날개패널들이 붙어있는 구조로 설계되었으며, 몸통의 끝에는 부채꼴 모양의 꼬리날개를 가지고 있다. 각 팔은 몸통과 3

자유도의 조인트로 연결되어 있으며, 꼬리는 pitch와 roll의 2자유도 움직임을 가진다.

날개의 각 패널들은 그에 가해진 외력에 대해 순응적으로 굽어질 수 있는 각스프링으로 이루어진 조인트로 팔에 연결되어 있다. 비록 모델링상으로는 각각의 패널들이 독립적으로 움직임으로 인해 패널 사이에 공간이 벌어져 실제의 공기역학적인 관점에서 볼 때 부정확한 면이 있으나, 이는 물리적모델의 단순화의 과정으로서, 실제 연속체인 날개가 무한개의 극소너비의 패널들로 표현될 수 있다는 것을 감안할 때 이를 선형적으로 단순화시킨 모델이라고 할 수 있다. 제한된 계산능력으로 인하여 이산화된 요소들로 구현되었으나 본 논문에서 표현하고자 하는 날개의 순응성과 그에 따른 외력의 감지를 통합할 수 있는 최소한의 설정이라고 할 수 있다. 이러한 날개구조는 가상의 "깃털"에 대한 모델로 볼 수 있으며, 형태계산 개념을 날갯짓 로봇에 적용하면서도 계산량을 줄이기 위해 고안되었다. 이러한 유연한 패널들은 불규칙할 수 있는 외부 힘에 대해 공기역학적 쿠션으로 작용함으로써 날갯짓 제어의 경직성을 줄여주는데, 이는 날개의 제어신호가 완벽하게 정확하지 않더라도 물리적인 순응성이 그 차이로 인한 불안정성을 흡수할 수 있게 된다. 예를 들어 공기 중에서는 약간의 활개침의 차이로 인한 stall이나 비대칭 모멘트등 매우 극적인 공기항력의 차이가 생길 수 있어 비행의 안정성에 큰 영향을 줄 수 있는데, 이러한 날개의 유연성은 각 날개패널들이 순응적으로 휘어짐으로써 제어 노이즈나 불규칙한 외력으로 인한 불안정성을 크게 줄일 수 있게 된다. 또한, 실제 깃털을 가진 새에서는 딱딱한 비행기의 날개보다 넓은 angle of attack범위에서도 날개주위로 이동하는 공기 흐름이 laminar하도록 유지될 수 있는데[15], 이는 갑작스럽게 양력을 잃어버리는 stall현상을 줄이면서도 보다 많은 양력과 추력을 가능하게 한다.

이러한 유연한 패널 조인트에 위치한 센서들은 로봇의 날개에 가해지는 항력을 감지할 수 있도록 하는데, 이는 실제 새들이 날개의 휘어짐과 깃털의 떨림등의 감각으로 공기의 흐름 감지할 수 있다는 점과 유사하다[16]. 깃털과 관련된 mechanosensor 신호를 분석한 연구에 따르면 깃털 및 날개에 분포된 감각들은 stall의 감지와 더불어 날개 상에서 공기흐름의 분리지점 및 날개의 부드러운 2차 깃털의 떨림수를 통하여 현재의 공기속도를 감지할 수 있음이 밝혀져 있다[17,18]. 이는 곤충의 털과 더듬이와 유사하게,

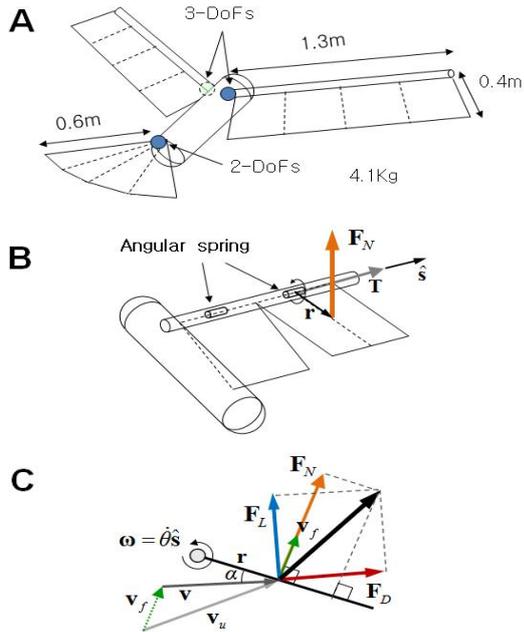


Fig. 1. Ornithopter robot and aerodynamics model

새에서도 날개와 깃털로 감지하는 여러 힘에 대한 정보가 날갯짓 운동에 관여할 수 있다는 가능성을 제시한다. 본 논문에서의 날개의 감각은 걷는 로봇에서의 접촉, 압력 센서와 유사하게 활용되었는데, 이는 걷는 로봇이 지면의 용기와 자신의 무게를 발 센서로 감지하듯이 본 날개패널 센서 모델 역시 주변을 둘러싼 유체로부터 가해지는 압력을 감지하는 것이라 생각 할 수 있다.

Fig. 1A는 본 연구의 날갯짓 로봇 모델을 보여주고 있다. 로봇의 총 질량은 4.1Kg이며 날개양끝 너비 (wing span)는 2.6m, aspect ratio는 6, wing loading은 0.35g/cm²이다. 날개팔의 3자유도 관절은 각각 dihedral, sweep, twist를 나타내며 각 관절 가동범위는 dihedral축으로 ±45°, sweep과 twist는 ±22.5로 설정되었다. 두 날개의 고정 각도는 위쪽으로 22.5° 기울어져 있어서 공기중에서의 중력방향에 대한 static 안정성을 높이게 하였다[19]. 꼬리날개를 위한 관절은 pitch와 roll 모두 ±22.5°의 가동범위를 가지며 꼬리의 펼쳐짐 (spread)은 45°로 고정되었다. 각 날개 패널들은 그 무게중심에 양력 (lift)과 항력 (drag)이 작용하도록 계산되며 Thin airfoil 및 Blade element theory를 바탕으로 한 Quasi-steady 가정으로

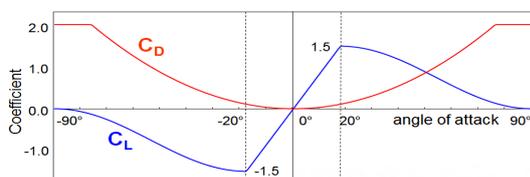


Fig. 2. Aerodynamic coefficients

단순화된 공기역학 모델[20]이 사용되었다. 날개패널의 leading edge로 유속 $v=|v|$ 의 공기흐름이 α (rad)의 angle of attack으로 들어온다고 가정했을 때 양력과 항력 벡터 들은 다음과 같이 계산된다.

$$F_L = \frac{1}{2} C_L(\alpha) \rho A |v| v$$

$$F_D = \frac{1}{2} C_D(\alpha) \rho A |v| v$$

이 때 ρ (1.21kg/m³)는 공기의 밀도, A는 패널의 넓이다. 움직이고 있는 날개패널이 바라보는 공기의 흐름은 로봇으로 들어오는 본래의 흐름 v_u 와 패널 중심점의 속도 v_f 의 차이로 구해진다(Fig 1C). 매 순간 패널중심점의 속도는 팔의 선형속도와 각속도로 인한 패널중심점의 속도에 더하여, 날개패널이 패널조인트를 중심으로 회전하는 각속도와 회전점에서 패널 중심점까지의 벡터 r 의 외적으로 구할 수 있다. Angle of attack (α)의 함수로 표현되는 양력과 항력계수는 Fig 2와 같이 piecewise continuous한 함수로 모델링되었다(그 외 자세한 사항은 [4-6]을 참조).

날개패널의 회전을 위한 각스프링은 패널의 무게가 없다고 가정한 후 일차 미분방정식으로 표현되었다. 어떤 초기 각도에서 시작할 때 패널 조인트의 각도 θ 는 그 순간의 각스프링 모멘트와 패널에 가해진 힘으로 인한 모멘트가 균형을 이루는 지점으로 부드럽게 수렴하도록 구현되었다. 매 순간마다 패널스프링 θ 의 각도와 각속도는 다음과 같은 식을 통해 업데이트 된다.

$$\frac{d\theta_i}{dt} = P(T_i - k\theta_i)$$

T_i 는 외부로부터 가해진 모멘트이고, k 는 스프링 상수 (0.1Nm/rad), 그리고 $P=100$ 는 비례상수이다. 패널에 가해진 회전모멘트는 패널이 받는 전체 공기역학적 힘의 수직성분 F_N 과 힘이 가해지는 지점까지의 모멘트암 (r)으로 구해진다 (Fig 1B). 패널회전을 제외한 로봇의 시뮬레이션은 Open Dynamics Engine[21] 라이브러리를 사용하여 구현되었으며 모든 시뮬레이션은 time step 0.01sec로 업데이트 되었다.

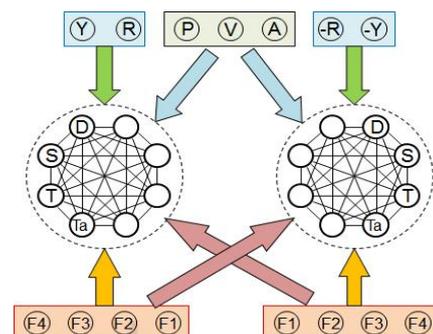


Fig. 3. Neural circuit

III. Neural Circuit

날갯짓 비행 컨트롤을 위한 신경망은 Continuous time Recurrent Neural Network (CTRNN) 모델[22]이 사용되었다. 신경망의 연결구조는 Fig 3과 같이 대칭적 구조로 설계되어 좌우 양쪽의 컨트롤을 같게 함으로써 양날개 제어에 필요한 파라미터 수를 반으로 줄일 수 있다. 그림에서 각각의 색으로 나타내어진 그룹들은 서로 거울 대칭을 이루는 부분으로서 동일한 파라미터 세트를 갖는다. 한쪽 날개당 8개 neuron으로 된 fully connected 망을 사용하며 그 중 4개의 neuron이 각 관절모터를 제어한다 (D: dihedral, S: Sweep, T: Twist, Ta: Tail). 꼬리는 양쪽의 Ta neuron들 (Right/Left)로부터 pitch에 대해 $(R+L)/2$, twist에 대해 $(R-L)/2$ 의 신호로 제어된다. 신경망의 각 neuron은 다음의 식에 의해 업데이트 된다.

$$\tau_i \frac{dy_i}{dt} = -y_i + \sum_{j=1}^N \{w_{ji} \sigma(y_j + b_j)\} + I_i$$

위 식에서 y_i 는 neuron의 상태값이며, τ_i 는 time constant, w_{ji} 는 neuron j 로부터 i 로의 연결강도이다. b_j 는 bias이고, I_i 는 외부입력을 나타낸다. $\sigma(x)=\tanh(x)$ 는 neuron의 activation 함수이다. 파라미터 탐색을 위한 범위값은 연결강도와 bias가 $[-4,4]$ 이고, time constant는 $[0.01,1]$ 의 범위를 갖도록 설정되었다. 모든 신경망 노드들은 forward Euler 수치적분법을 사용하여 time step 0.01sec로 시뮬레이션 되었다. 전체 신경망의 구조는 Fig 3과 같이 자기 자신과의 연결을 포함한 fully recurrent 구조이며 모든 sensor neuron들로부터 대칭적인 연결을 받는다. Sensor neuron들은 해당 센서로부터 직접 값을 받으며 오직 $[-1,1]$ 범위의 bias값만을 갖는다. 그림에서 P,Y,R은 각각 로봇 몸통의 pitch, yaw, roll 정보를 입력 받으며 두 개의 RNN으로 거울대칭적으로 입력된다. pitch와 roll은 중력방향 벡터와의 각도로 결정되며, yaw는 목표지점에서의 방향각도로 주어졌다. A와 V는 각각 목표위치를 기준으로 한 몸통의 고도(altitude)와 비행속도값을 받게 된다. 모두 8개의 날개패널로부터의 굽힘센서

정보를 받으며, 순간적인 외력에 대한 민감도를 고려하여 패널조인트의 각속도값을 입력으로 사용하였다. 두 그룹의 RNN 사이에는 양날개의 불필요한 out-of-phase 동기화를 방지하기 위하여 연결이 없도록 하였다. 하지만 반대쪽 날개의 패널센서들의 값을 받도록 함으로써 서로 간접적으로 정보를 주고받을 수 있는 구조로 설계되었다.

IV. Evolutionary Search

신경망 연결강도 최적화를 위하여 지리적 분산 진화 알고리즘 (Geographically distributed evolutionary algorithm) [23]이 사용되었다. 이는 각 개체 (individual)들이 오직 자신과 가까운 지리적 이웃들과만 상호작용하도록 함으로써 local optimum 문제를 완화하기 위한 방식으로, 실제 진화과정에서의 지리적 niche 개념을 도입한 진화연산의 한 종류이다. 본 논문에서는 개체군 크기가 400으로 20×20 격자로 이루어진 torus 공간을 사용하였다. 한 개체가 위치한 격자 주변의 8개 개체를 1차 이웃으로, 그 바깥의 16개 개체를 2차 이웃으로 하여 그 범위 안에서 거리에 따라 교배선택 확률을 지정하였다. 각 개체를 나타내기 위한 gene encoding은 $[-10000,10000]$ 범위의 integer의 배열로 설정하여 각각의 신경망 파라미터가 갖는 범위로 스케일링 하였다. 자식개체를 위한 선택은 Tournament selection을 사용하였으며, 1에서 3 point crossover를 각각 1/3의 확률로 사용하였다. Crossover 후 각 개체의 adaptive mutation rate에 따라 돌연변이가 적용되었다. 초기에 모든 개체는 50%의 돌연변이율을 가지고 시작한 후, 자식의 점수가 두 부모의 평균 점수의 70% 이상이면 두 부모의 돌연변이율을 0.1% 증가시키고, 그렇지 않으면 감소시켰다 (더욱 자세한 사항은 [5] 참조).

1. Fitness Evaluation

이전 연구들은 방향전환이 가능한 날갯짓 로봇 컨트롤을 진화시키기 위한 방법으로 먼저 직선비행 후 꼬리제어를 중

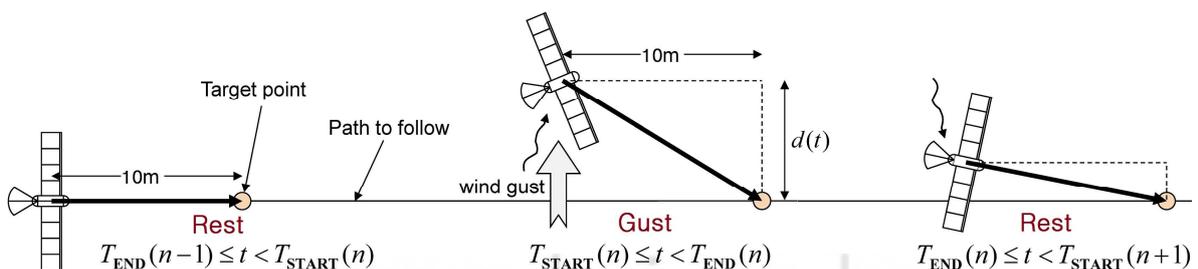


Fig. 4. Evaluation strategy

심으로 방향전환을 추가하는 단계별 진화 방법을 사용하였으나 본 연구는 보다 전략적으로 설계된 새로운 단일평가방법을 사용하여, 견거나 헤엄치는 로봇의 경우와 같이 한 번의 통합된 진화연산으로 기동성과 안정성 두 목표를 동시에 고려하고자 한다. 이 때, 대부분의 견거나 헤엄치는 사지기동 로봇의 경우와 다르게, 날갯짓 기동을 위한 적합도 평가는 그 특유의 역학적 불안정함으로 인하여 보다 세심한 설계를 필요로 한다. 예를들어 평가 시뮬레이션을 공중에서 시작할 경우에, 양 날개의 작은 stroke 차이로 인한 역학적 불균형은 대부분의 개체가 바닥으로 추락하는 결과로 이어지기 쉽다. 이처럼 초기에 대부분의 개체가 최저치의 적합도를 가지게 되어 진화연산을 위한 변별력이 사라지는 bootstrap problem이 본 시스템에서는 특히 두드러진다고 할 수 있다. 이러한 문제를 조금이라도 완화하기 위하여, 본 연구는 목표 지점을 일정 고도에 위치한 후, 지상에 놓인 로봇을 정해진 초기속도를 가지도록 앞으로 밀어주는 방식으로 시뮬레이션을 시작하였다. 그럼으로써 한 개체가 성공적으로 이륙하기 위해서는 자연스럽게 날갯짓 비행능력과 목표추적 능력을 동시에 만족 시켜야 가능하도록 평가가 이루어진다. 로봇은 마찰이 없는 지상에 놓여진 후 전방으로 초기속도 7m/s로 밀어지도록 하였으며, 시작순간에 급격한 팔의 움직임으로 인한 불안정성을 방지하기 위해 1초가 지난 후 공기항력이 적용되었다. 이륙하지 못하거나 잠시 후 추락하는 개체들에 대한 불필요한 시뮬레이션을 가려내기 위하여, 시뮬레이션 시작 후 5초 후에도 몸체의 고도가 0.5m 이하인 경우엔 평가를 종료하였다. 그 이후에는 고도 0.3m 이하가 되면 추락으로 가정하여 시뮬레이션을 역시 종료하였다.

총 60초의 시뮬레이션 시간동안 로봇은 고도 10m 공중에 고정된 전방으로의 직진경로를 유지하여야 하는데, 목표지점은 항상 현재의 로봇 위치로부터 전방 10m 앞에 위치하게 하여 실제 로봇이 보는 목표는 로봇의 전방 10m 앞에 있는 평면에 투영된 위치를 보게 된다. 이를 통해 로봇의 yaw각도와 목표기준 고도가 측정되어 센서입력으로 주어진다. 목표지점은 측면이나 위아래로 움직이지 않으며, 대신 임의의 순

간에 임의의 강도와 세기를 가진 바람을 이용한 섭동(Gust perturbation)을 주었다. 이는 이륙에 성공한 개체들에 대해 목표로의 방향전환 능력과 외부 섭동에 대한 안정성을 동시에 통합적으로 평가하는 매우 효과적인 방법으로 고안되었는데, 이를 통해 로봇은 외력이 주어지는 상황 하에서 목표지점까지의 거리를 최소화하여야만 높은 평가를 얻게 된다. Gust perturbation은 시작 5초 후부터 랜덤하게 주어지게 하여, 그 이전에는 단지 목표고도까지의 이륙을 성공해야 하도록 설계되었다. 비행로봇의 매우 비선형적이고 불안정한 역학적 상황과 좁은 날갯짓 각도의 허용치 등의 내재된 어려움으로 인하여, 각 개체는 이러한 단순해보이는 섭동 전략에도 불구하고 다양하고 민첩한 날개 컨트롤 전략을 진화시켜야 살아남을 수 있게 된다. 로봇개체의 평가를 위한 Fitness function은 다음과 같이 표현된다.

$$\text{Fitness} = 100\{pf_1 + (1-p)f_2\}, \quad p = 0.5f_2$$

$$f_1 = \frac{1}{1+d}, \quad f_2 = \frac{T}{T_{\max}} \frac{A}{A_{\min}} \frac{F}{F_{\min}}$$

여기서 T 는 시뮬레이션 종료까지의 시간, T_{\max} 는 최대 시뮬레이션 시간(60sec), d 는 로봇시야 평면상에서의 목표 지점까지의 평균거리, A 와 F 는 시뮬레이션 시간동안 평균 날갯짓 amplitude와 frequency이다. 날갯짓 진폭과 주파수는 신경망 컨트롤러의 반복적인 날갯짓 신호의 존재여부를 강제하기 위함이며, 각각의 최소요구값($A_{\min}=0.6\text{rad}$, $F_{\min}=2\text{Hz}$)만을 만족하면 되도록 설정하였다 ($A/A_{\min} \leq 1$, $F/F_{\min} \leq 1$ 로 제한). 두 개의 개별 fitness의 가중치 p 는 f_2 에 대해서 점진적으로 변화하도록 설정하였는데, 최대 시뮬레이션 시간동안 추락하지 않고 요구된 날갯짓을 유지할 때 비로소 목표추적에 대한 점수와 1:1의 평가 가중치를 가지도록 하였다. 예를 들어, 로봇이 일찍 추락할수록 최종 적합도는 목표점에서의 기동보다 날갯짓 생성에 더욱 많은 가중치를 두게 된다. 이러한 adaptive weighting 전략은 단일 평가에 있어서 서로 상충하는 목표들인 안정적 비행과 목표방향으로의 기동을 균형있게 조율하는 역할을 하도록 한다.

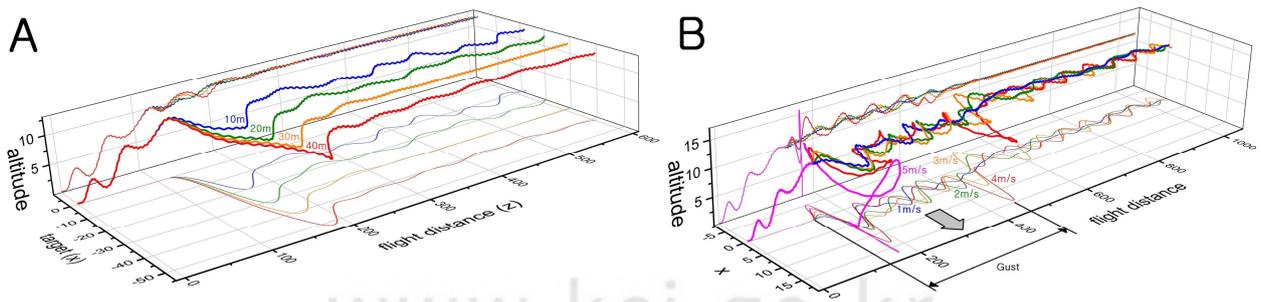


Fig. 5. Flight trajectories for: (A) steering toward target, (B) reaction for gust perturbation.

V. Experiment and Result

총 60초의 주어진 적합도 평가 시뮬레이션 시간동안 수 차례의 gust perturbation이 주어진 범위 내에서 각각 랜덤한 시간동안 랜덤한 방향과 세기로 가해졌다 (Fig 4). 각각의 gust session (n)은 시뮬레이션 시작 5초 후부터 gust의 시작시점 ($T_{START}(n) \leq t$)을 후, 1-5m/s의 gust를 1-3초간 가하게 된다 ($t < T_{END}(n)$). 한번의 gust session이 후에는 다시 1-3초간의 no-perturbation 시간이 주어짐으로써 바람에 의해 밀려나가거나 불안정해진 로봇이 다시 원래의 코스로 돌아올 시간을 주게 된다. 이러한 방법으로 전체 시뮬레이션 시간 동안 수 차례의 gust perturbation과 휴지시간을 랜덤한 interval로 번갈아가며 경험하게 된다. 바람의 방향은 이전 session과 유사하지 않도록 이전 방향으로부터 로봇 몸통을 중심으로 하는 구면좌표상으로 30도 이상 차이가 나도록 설정하였다.

진화연산은 평균 총 50000에서 70000번의 개체평가로 진행되었으며 3.4Ghz 단일 CPU에서 평균 8~10시간이 소요되었다. 성공적으로 진화된 날개로봇들은 평균 1-2Hz의 wingbeat frequency를 가지고 9-12m/s의 비행속도를 보여주었는데, 이는 설계된 로봇과 크기와 무게가 비슷한 대형 독수리의 평균 순항속도에 비해 약간 낮으나 대체로 비슷한 수준으로서, 날개크기와 무게와의 관계가 중요한 조류비행의 기준으로 볼 때 본 시스템의 역학적 모델과 진화연산이 타당한 움직임을 찾아내었다고 볼 수 있다. 단순화된 가정을 이용한 로봇과 공기역학 시뮬레이션이지만 진화된 로봇들은 상당히 자연스럽게 안정적인 날갯짓 기동을 보이면서도 목표지점을 향해 steering 기동을 보여주었다(Fig 5A). 비행중 사용자가 수동으로 gust를 주었을 때 역시 비행속도의 50% 이상까지의 유속에 의한 섭동 (~5m/s)에도 추락하지 않고 비행을 유지할 수 있었다(Fig 5B). 측면으로의 섭동이 주어졌을 때 로봇은 바람이 불어오는 방향과 목표지점 사이의 방향으로 자세를 틀었는데

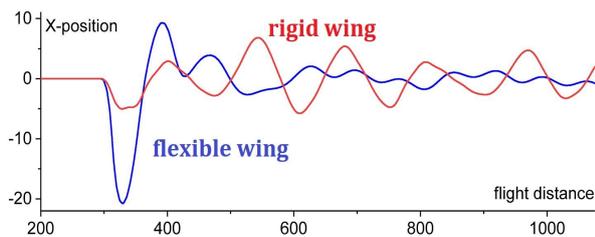


Fig. 6. Trajectories of panel-wing vs rigid wing after perturbation

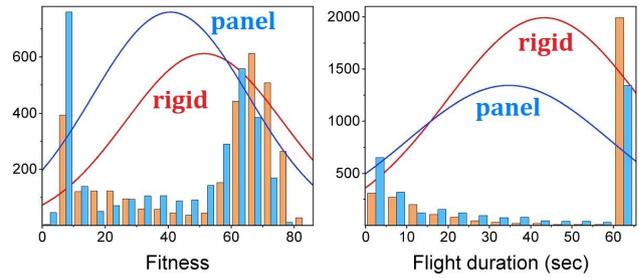


Fig. 7. Histograms for panel-wing vs. rigid wing

이는 날갯짓 공기역학으로 인한 수동적 현상이 더 큰 영향을 미친 것으로서, 약간 전진익 형태가 되었을 때의 측면으로의 바람은 바람에 먼 쪽의 날개에 더 큰 추력을 주게 되어 전체적으로 upstream방향을 향해 자연스럽게 몸을 틀게 된다. 이 역시 진화연산이 찾아낸 또 다른 생존전략이라고 할 수 있겠다. 유연한 날개패널의 순응적 안정성 이외에 패널센서도 날갯짓 패턴에 관여함을 알 수 있는데, 만약 패널센서를 차단하면 메인 신경망에서는 oscillatory 패턴이 사라지거나 또는 oscillatory output을 생성하지만 패널센서의 부재로 날개의 움직임이 작은차이로 변화되어 결국 날지 못하는 결과를 볼 수 있었다. 패널센서를 유지한 채 타겟센서를 제거한 후 gust를 가해도 역학적 안정성과 함께 pitch, roll, 패널센싱을 통하여 추락하지 않고 비행을 유지할 수 있었다. 소수의 특별한 개체에서는 목표타겟이 로봇보다 아래에 있을 때 날갯짓을 멈추고 활강하는 gliding 움직임을 보여주었다 (자세한 로봇의 움직임은 첨부된 동영상을 참조).

날갯짓 비행에 있어서 유연한 날개는 단단한 날개보다 더 나은 lift-to-drag ratio를 보인다고 알려져 있다. 그러나 일반적인 여러 상황에서의 두 상황의 날갯짓 비행을 비교하기는 쉽지 않은데, 이는 날개의 물리적 특성에 따라 항상 날갯짓의 kinematics와 비행속도 등이 달라지기 때문이다[19]. 간단한 두 경우를 비교를 위해 새로이 패널의 움직임이 고정된 rigid wing를 사용하여 로봇을 진화시킨 후 비교를 진행하였다. 단단한 날개로봇의 경우 모든 패널 센서들이 제외되므로 더 적은 파라미터 탐색이라는 이점이 있었으며, 역시 성공적으로 진화 후에 좋은 비행을 보여주었다. 그러나 gust 섭동이 주어졌을 때 원래 코스로 돌아가는 움직임이 더욱 불안정하고 느린 것을 확인할 수 있었다(Fig 6). 추가로 두 경우 로봇의 진화연산의 과정을 관찰해보기 위해 두 로봇을 steering 기동 없이 일정한 고도를 유지하며 날도록 진화시키며 새로운 개체가 평가될 때마다 그의 적합도와 비행시간을 관찰한 결과, 단단한 날개를 가진 쪽의 적합도와 비행시간 분포가 더 양극화되어

있음을 확인할 수 있었다(Fig 7). 이는 두 종류 로봇 모두 날갯짓 비행을 할 수 있는 신호패턴을 생성할 수 있지만, 단단한 날개로봇의 신호패턴의 다양성이 유연한 날개로봇보다 떨어진다는 것을 보여주는데, 다시 말해 단단한 날개를 가진 로봇은 유연한 패털날개보다 더 정확한 날갯짓 움직임 신호를 최적화로 찾아내어야만 추락하지 않고 비행할 수 있다는 것을 의미한다. 추가로 두 군집의 gene의 다양성을 측정하기 위해 두 개체군의 Nearest neighbor distance 측정[24]($D_{NN} = \Sigma_i \{\min(d_{ij})/N\}$, $i \neq j$)을 계산해보면 유연한 날개의 개체군은 0.422, 그리고 단단한 날개 개체군은 0.407로 유연한 날개 개체군쪽이 더 높은 유전적 다양성을 가짐을 확인할 수 있었다.

VI. Discussion

본 연구를 단계적인 최적화 방법을 사용하지 않고도 한번의 통합적 진화연산으로 steering기동과 외력에 의한 안정성 두가지를 모두 만족하는 날갯짓 로봇을 위한 진화연산 방법을 제시하였다. 비록 제안된 시스템이 상당한 범위 내의 섭동에 대해 안정적인 기동을 보여주었으나, 향후 실제 상황과의 reality gap을 줄이기 위해 추가적인 연구가 필요하다. 첫째로, 단순화된 날개패널의 독립적 움직임으로 모델링된 날개 시뮬레이션은 본 연구의 개념적 타당성을 설명할 수 있으나, 실제의 연속적인 날개유연성을 위하여 FEM등의 연속체 모델로 개선될 필요가 있다. 수만번의 평가 시뮬레이션이 필요한 진화연산의 특성상, 시뮬레이션의 정확도와 계산시간의 조율이 필요하며, 이는 알고리즘 성능의 개선과 강력한 계산성능의 도입을 통하여 개선해나가야 하겠다. 두 번째로는 더욱 많은 자유도를 가진 실제 새와 유사한 관절을 사용한 로봇 시뮬레이션을 위해 보다 큰 스케일의 RNN을 최적화할 필요가 있다. 이 역시 시뮬레이션의 정확도와 효율적인 분산컴퓨팅 그리고 계산성능의 증가가 필수적이라고 하겠다. 로봇 시뮬레이션과 신경망의 구성과 독립적으로, 본 연구의 진화연산 평가전략을 통해 어떤 모델이라도 효과적으로 성공적인 비행을 생성해 낼 수 있을 거라 예상된다. 가중치로 묶인 한 개의 적합도 함수 대신 Multi-objective 진화연산을 사용하여 steering과 안정성, 에너지 소모 등을 최적화시키기 위한 후속연구도 필요하다. 이러한 경우 gliding과 soaring, intermittent flapping flight 등 더욱 실제 조류와 같은 움직임들 뿐 아니라 그러한 움직임들 간의 상황에 따른 지능적인 전환 등이 가능할 것으로 기대된다.

REFERENCES

- [1] D.D. Chin, and D. Lentink, "Flapping Wing Aerodynamics: From Insects to Vertebrates," *Journal of Experimental Biology*, Vol. 219, pp. 999-999, 2016.
- [2] C. Chen, and T. Zhang, "A Review of Design and Fabrication of the Bionic Flapping Wing Micro Air Vehicles," *Micromachines*, Vol. 10, No. 144, pp. 1-20, 2019.
- [3] J. Gerdes, H.A. Bruck, S.K. Gupta, "A Simulation-based Approach to Modeling Component Interactions during Design of Flapping Wing Aerial Vehicles," *International Journal of Micro Air Vehicles*, Vol. 11, 2019.
- [4] J. Wu, and Z. Popović, "Realistic Modeling of Bird Flight Animations," *ACM Transactions on Graphics*, Vol. 22, No. 3, pp. 888-895, 2003.
- [5] Y.S. Shim, and C.H. Kim, "Evolving Physically Simulated Flying Creatures for Efficient Cruising," *Artificial Life*, Vol. 12, No. 4, pp. 561-591, 2006.
- [6] Y.S. Shim, S.J. Kim, C.H. Kim, "Evolving Flying Creatures with Path-following Behavior," *The 9th International Conference on the Simulation and Synthesis of Living Systems (ALIFE IX)*, Boston, MA, pp. 125-132, 2004.
- [7] J. Won, J. Park, K. Kim, J. Lee, "How to Train Your Dragon: Example-Guided Control of Flapping Flight," *ACM Transactions on Graphics*, Vol. 36, No. 4, 2017.
- [8] J. Won, J. Park, J. Lee, "Aerobatics Control of Flying Creatures via Self-Regulated Learning," *ACM Transactions on Graphics*, Vol. 37, No. 6, 2018.
- [9] F.P. Such, V. Madhavan, E. Conti, J. Lehman, K.O. Stanley, J. Clune, "Deep Neuroevolution: Genetic Algorithms are a Competitive Alternative for Training Deep Neural Networks for Reinforcement Learning," *NIPS Deep Reinforcement Learning Workshop*, 2018.
- [10] K.O. Stanley, J. Clune, J. Lehman, R. Miikkulainen, "Designing Neural Networks through Neuroevolution," *Nature Machine Intelligence*, Vol. 1, No. 1, pp. 24-35, 2019.
- [11] J.B. Mouret, S. Doncieux, J.A. Meyer, "Incremental Evolution of Target-following Neuro-controllers for Flapping-wing Animats," *SAB 06, LNCS (LNAI)*, Vol. 4095, pp. 606-618, 2006.
- [12] E. de Margerie, J.B. Mouret, S. Doncieux, J.A. Meyer, "Artificial evolution of the morphology and kinematics in a flapping-wing mini UAV," *Bioinspiration & Biomimetics*, Vol. 2, pp. 65-82, 2007.
- [13] D.R. Warrick, M.W. Bundle, K.P. Dial, "Bird Maneuvering Flight: Blurred Bodies, Clear Heads," *Integrative and Comparative Biology*, Vol. 41, No. 1, pp. 141-148, 2002.
- [14] R. Pfeifer, and F. Iida, "Morphological Computation: Connecting Body, Brain and Environment," *Japanese Scientific Monthly*, Vol.

- 58, pp. 48-54, 2005.
- [15] E.H. Burt Jr., and J.M. Ichida, "Selection for Feather Structure," *Acta Zoologica Sinica*, Vol. 52, pp. 131-135, 2006.
- [16] M. Gewecke, and M. Woike, "Breast Feathers as an Air-current Sense Organ for the Control of Flight Behaviour in a Songbird (*Carduelis Spinus*)," *Zeitschrift für Tierpsychologie*, Vol. 47, No. 3, pp. 293-298, 1978.
- [17] R. Necker, "Somatosensory System," *Physiology and Behavior of the Pigeon*, pp. 169-192. Academic Press, London (1983)
- [18] R.E. Brown, and M.R. Fedde, "Airflow Sensors in the Avian Wing," *Journal of Experimental Biology*, Vol. 179, No. 1, pp. 13-30, 1993.
- [19] A.L.R. Thomas, and G.K. Taylor, "Animal Flight Dynamics I. Stability in Gliding Flight," *Journal of Theoretical Biology*, Vol. 212, No. 3, pp. 399-424, 2001.
- [20] T. Weis-Fogh, and M. Jensen, "Biology and Physics of Locust Flight I: Basic Principles in Insect Flight. A Critical Review," *Philosophical Transactions of the Royal Society of London B*, Vol. 239, No. 667, pp. 415-458, 1956.
- [21] R. Smith, "Intelligent Motion Control with an Artificial Cerebellum," *Doctoral Dissertation, University of Auckland, New Zealand, 1998*, <http://ode.org/>
- [22] R.D. Beer, "On the Dynamics of Small Continuous-Time Recurrent Neural Networks," *Adaptive Behavior*, Vol. 3, No. 4, pp. 469-509, 1995.
- [23] P. Husbands, "Distributed Coevolutionary Genetic Algorithms for Multi-criteria and Multi-constraint Optimisation," *Evolutionary Computing. LNCS*, Vol. 865, pp. 150-165, 1994.
- [24] P.J. Clark, and F.C. Evans, "Distance to Nearest Neighbor as a Measure of Spatial Relationship in Populations," *Ecology*, Vol. 35, No. 4, pp. 445-453, 1954.

Authors



Yoonsik Shim received the B.S. in Mechanical Engineering and M.S. in Computer Science from Korea University, Korea. He received the Ph.D. degree in Informatics from University of Sussex, UK, in 2013. Dr. Shim

is currently a Research Professor in Institute of Computer, Information and Communication, Korea University. His research interests are focused on bio-inspired adaptive robotics, computational neuroscience, chaotic neurodynamics, self-organisation, and evolutionary robotics.