

From Wheels to Wings

with Evolutionary Spiking Circuits

Dario Floreano¹, Jean-Christophe Zufferey^{1,2}, Jean-Daniel Nicoud²

¹Autonomous Systems Lab, Institute of Systems Engineering
Swiss Federal Institute of Technology (EPFL), CH-1015 Lausanne, Switzerland (<http://asl.epfl.ch>)

²DIDEL S.A., CH-1092 Belmont/Lausanne, Switzerland (<http://www.didel.com>)

Abstract. We give an overview of the EPFL indoor flying project, whose goal is to evolve neural controllers for autonomous, adaptive, indoor micro-flyers. Indoor flight is still a challenge because it requires miniaturization, energy efficiency, and control of non-linear flight dynamics. This ongoing project consists in developing a flying, vision-based micro-robot, a bio-inspired controller composed of adaptive spiking neurons directly mapped into digital micro-controllers, and a method to evolve such a neural controller without human intervention. This document describes the motivation and methodology used to reach our goal as well as the results of a number of preliminary experiments on vision-based wheeled and flying robots.

1. Introduction

Most research in embodied intelligence is carried out on terrestrial robots with wheels or legs. These robots are quite convenient because they move on a plane, are easily available on the market, and can be quickly assembled with off-the-shelf components. Terrestrial robots let researchers concentrate more on theoretical and algorithmic issues than on energetic autonomy.

Intelligent life on this planet evolved equally well on the ground, in the water, and in the air. In particular, airborne organisms have been very successful because wings allowed them to move across comparatively larger areas than terrestrial and aquatic animals, and thus colonize new environments. The greatest part of flying organisms is represented by insects, which are among the oldest animals on the planet (350 million years). Although they use a large variety of wings and actuation mechanisms, the low density of air forced flying organisms to develop morphological, bio-mechanical, sensory, and neural structures that balance size, weight, and energetic consumption. Insects, in particular, feature comparatively smaller size than any other flying animal. Nonetheless, insects exploit complex visual information to fly in cluttered environments and efficient adaptive neural circuits to map sensory information into motor actions necessary to stay airborne, control their body orientation and three-dimensional trajectory, avoid hitting obstacles, compensate for turbulences, aerodynamic effects, and inertial forces.

Today, there are not yet autonomous flying robots capable of maneuvering in small cluttered environments as insects do. To meet that challenge, at EPFL we started a project to develop

autonomous, vision-based, indoor flying robots. Most algorithms and technology used on terrestrial robots cannot match size, weight, and energy consumption requirements of micro-flying robots. In this project, we focus on adaptive neuro-morphic circuits and hardware miniaturization, but do not yet attempt to reproduce the bio-mechanical principles of insect flight. Our short-term goal is to develop a methodology that allows very small winged robots (50-80 cm wingspan) to fly autonomously within a room using only visual information. To achieve that goal, we build upon low-power digital micro-electronics, miniature CMOS micro-cameras, wireless communication, rechargeable low-weight batteries, and on an incremental approach through wheeled, lighter-than-air, and winged robots.

In this paper we provide an overview of the project with special emphasis on the methodology and technology used to provide these robots with onboard, adaptive intelligence. After a short survey of related literature in small flying robots and biologically-motivated vision-based navigation, we describe our choice of evolutionary spiking neural circuits to control vision-based small robots. We then describe an incremental series of experiments where we gradually take this methodology from miniature wheeled robots to micro-robots, to airships, and finally to winged robots. Since the latter stage is still in progress, we provide only a summary of the technical issues and first prototypes.

2. Related Work

2.1 Towards Miniature Autonomous Flying Robots

Flying in a sitting room is probably more difficult than flying in open sky because the space is small, there may be several obstacles of different shape and texture, and illumination may vary quite strongly within a few meters and come from several directions. Today, there are not yet flying vehicles capable of autonomously navigating inside a house.

Insects are very good at flying within rooms and represent therefore a rich source of inspiration for researchers who attempt to build small flying robots. Ron Fearing's team at Berkeley University is attempting to create a micro, flying robot that replicates wing mechanics and dynamics of flies. A piezoelectric actuator is used for flapping and rotating two 10 mm wings at 150 Hz. Electrical power is planned to be supplied by lithium batteries charged by three miniature solar panels [4]. So far, a single 2 cm² wing on a test stand has generated some lift while linked to an off-board power supply. A team at Stanford University proposed a centimeter scale rotorcraft named Mesicopter [17]. It is based on four miniature motors with 15 mm propellers. Experiments on lift and stability are done on larger models [18]. Yet another team at Caltech, in collaboration with Aerovironment, Inc. (<http://www.aerovironment.com>), developed the first radio-controlled, battery-powered, flapping-wing micro aerial vehicle with a 20 cm wingspan [26]. This 12-grams device performed a 6-minute remote-controlled flight with a Lithium-Polymer battery.

However, all the micro-mechatronic flying devices described above cannot yet navigate autonomously. In our project we aim at building micro flying devices that can carry

microelectronics, sensors, and batteries sufficient for several minutes of autonomous flight. We decided to use very small digital micro-controllers because such micro-controllers are largely available, well-tested, cheap, easily programmable, consume very little power, incorporate a number of devices useful for signal acquisition and, as we will show below, are suitable for very compact implementations of evolutionary spiking circuits to control the aircraft.

2.2 Bio-inspired Vision-based Navigation

Vision is a very rich source of information about the environment and, since it is a passive sensing device, it is more energy efficient than active sensors typically used in terrestrial robots, such as active infrared, sonar, and laser range-finder. Furthermore, the miniaturization trend driven by demand for multi-media consumer electronics is bringing to the market increasingly smaller and cheaper vision devices. For example, a commercial and fully packaged vision chip, composed of some thousands photoreceptors, with plastic optics can weigh less than 0.4 grams [11].

Vision systems of insects are mainly concerned with spatial and temporal change in the image. Spatial change is given by the relationship among activation values of adjacent photoreceptors measured within a single snapshot. Spatial relationship is useful for detecting contrast, shapes, and landmarks. Temporal change instead is given by the relationship among activation values of a single photoreceptor measured over time. Spatiotemporal relationship provides information about self-motion, motion of objects, and imminent collision.

In biological systems, spatiotemporal information is captured and mapped into motor actions by neuronal networks with architectures and time-dependent dynamics evolved to match ecologically relevant properties of the environment and of the interaction between the animal and the environment [13]. Scientists have been gradually unveiling the mechanisms of vision-guided behavior of insects by combining behavioral and neuro-physiological analysis with models and tests on terrestrial mobile robots. Nicolas Franceschini and his team at CNRS in Marseilles (for a review, see [10]) have spent several years to study the morphological and neurological aspects of the visual system of the fly and of its motion detection abilities. By means of a specially designed microscope, they were capable of correlating sequential stimulation of pairs of photoreceptors (equivalent to a moving spot) with the activity of single neurons within an inner assembly of neurons named lobula plata. The lobula plata features a set of topologically ordered neurons that receive mediated stimulation from photoreceptors on the compound eye of the insect. The authors speculated that the geometrical layout of photoreceptors and the connectivity pattern between photoreceptors and lobula plata neurons could provide a reliable estimation of distance from objects in ways that resemble the principles of motion parallax [38]. In order to test this hypothesis, they built an analog electronic circuit modeled upon the neural circuitry of the fly brain and interfaced it with a circular array of photoreceptors on a 12-kg wheeled robot. This vision-based robot was capable of approaching a goal while avoiding obstacles (characterized by higher contrast with respect to a background) on its way [9]. However, the system was still too heavy for autonomous flight and was based on a combination of straight motion and steering,

which may not be easily achieved in flying vehicles. The team is currently progressing on this area of research with tethered flying robots [29].

Heinrich Buelthoff and his team at Max-Planck Institute in Tuebingen approached the issue of insect vision-based navigation in two different ways. In a first set of experiments, they handcrafted an array of Elementary Motion Detection (EMD) neurons and connected them to the linear camera and motors of a wheeled Khepera robot. EMDs are neurons that extract motion directions by means of time delay between activations of adjacent photoreceptors [27]. They used an evolutionary algorithm to optimize the network parameters for a wheeled robot navigating in a variety of visual environments [14]. In another set of experiments, they took a closer look at the morphology and response properties of the fly compound eyes. In particular, they reconstructed a set of color intensity and motion vector fields by correlating visual input from the compound eyes of a simulated fly with typical actions taken by the insect in an open-sky environment. They showed that the resulting vector fields could be used to map visual information into suitable motor actions for a simple simulated insect hovering over a rugged, open-sky terrain [23]. These experiments were carried out in simulation because the hardware implementation was considered a nuisance rather than an advantage [24]. Although this latter issue is open to discussion [36], a major issue of that approach with respect to our goal is that it is designed to work in specific open-sky environments. Therefore, it is not applicable to indoor flight where there is no horizon, the agent is completely surrounded by texture, and the vertical light gradient is not as well defined as in natural environments. Indoor flight is likely to require more modifications than simply reconstructing the vector field for each indoor environment.

Mandyam Srinivasan and his team at the Australian National University in Canberra carried out an extensive set of experiments to understand visual performance of honeybees and tested the resulting models on vision-based wheeled robots (see [33], for a review). For example, they showed that honeybees regulate speed and direction of flight by balancing the speeds of image motion on the two eyes [34]. This mechanism was then demonstrated on a wheeled robot equipped with a camera that captured images of the lateral walls and transmitted them to a desktop computer where an algorithm attempted to balance the angular velocities of the images by steering the robot accordingly [37]. A number of other labs used this type of mechanism to drive wheeled robots in corridors (cited in [34]). In other work, the authors implemented an array of algorithms inspired upon insect flight to drive a vision-based robot in cluttered environments [31,32]. The algorithm related the position of the camera, the speed of the robot, and the measured velocities of images in order to judge distances from objects and steering accordingly. An interesting aspect of Srinivasan's work is that it shows that stereo vision and shape recognition are not necessary for insects and robots to navigate in cluttered environments and reach for goals. However, those systems require information about egomotion in order to reliably map optical flow into distances from obstacles. For this purpose, they either rely on odometry or use "saccadic movements" (straight movements during which distances are estimated followed by rapid turning actions without distance estimation). Saccadic movements are known to be used by insects, but are hardly applicable to aircrafts.

3. Evolutionary Spiking Circuits

The models described above attempt to replicate the morphology and/or neural architectures of specific flying insects. In this project we take a different approach consisting of exploring a class of biologically plausible neurons whose hardware implementation is suitable for the constraints of our indoor flying robots. We then use an evolutionary process to generate minimal, but functional, networks capable of vision-based navigation and are interested in comparing the resulting mechanisms with those used by insects to achieve similar performance. Unlike the above-mentioned projects, we neither specify a predefined trajectory, nor attempt to explicitly compute distance estimations from visual data. Instead, we use simple fitness functions to selectively reproduce the control systems of robots that can move forward as fast and as long as possible during their lifetime. Distance estimation as well as precise trajectories or behaviors are never explicitly specified.

There are two major classes of artificial neural networks whose dynamics are suitable for capturing spatial and temporal information required for vision-based navigation: Continuous Time Recurrent Neural Networks (CTRNNs) [1] and Spiking Neural Networks (SNNs) (for an introduction, see Maas and Bishop, 1998). Typically, in CTRNNs neuron states are characterized by time-dependent dynamics and continuous signals are propagated through the connections. Because of the potential complexity of these systems and difficulty to design them by hand, evolutionary methods have often been used to generate functional CTRNN [1,3]. For example, David Cliff used CTRNNs to successfully evolve obstacle avoidance behavior of a simulated hoverfly [2]. However, digital implementations of CTRNNs require relatively large memory to store floating-point variables such as time constants, weight strengths, thresholds, and look-up table for the non-linear activation function of the neurons.

Spiking neurons instead attempt to model the computational properties of pulsed neuronal networks. A pulse is a self-propagating perturbation of the electric voltage across the membrane of a neuron that occurs when the voltage exceeds a threshold. The pulse is also known as spike to indicate its short and transient nature. Spiking neurons have been mainly studied and formalized within the computational biology community [28], but are recently attracting an increasing interest by computer scientists and engineers [20] for their rich non-linear dynamics and compact hardware implementation. For example, it has been shown that large networks of spiking neurons can be implemented in tiny and low-power chips [15] exploiting the sub-threshold physics of transistors (that is, gate-to-source voltage differences below threshold voltage) in analog VLSI circuits [21]. Despite early promising implementations of hand-crafted spiking circuits for robot control [16,19], there are not yet methods for developing complex spiking circuits that could display minimally-cognitive functions or learn behavioral abilities through autonomous interaction with a physical environment. Furthermore, synaptic modification within analog VLSI circuits is still an open research and technology issue.

In this project, we investigate spiking neurons for vision-based navigation and address the above-mentioned challenges by exploiting a) artificial evolution in order to generate functional

networks and b) digital implementations in low-power micro-controllers that can be carried on-board by an indoor flying robot.

An evolutionary process is used to search through the connectivity space of networks of spiking neurons connected to an array of visual receptors (figure 1). All evolutionary experiments are entirely carried out on physical robots. Sensory receptors map activity levels of sensors into stochastic spike trains where the spike density is proportional to the activation of the corresponding sensor. The spike train of pairs of spiking neurons (working in push-pull mode) is mapped into speed values of corresponding actuators of the robots (wheels or propellers). The genetic representation is a binary string composed of a number of blocks, one for each neuron in the network. A block consists of one bit to encode the sign of the neuron (that is, the postsynaptic effect of outgoing spikes), n bits to encode the presence or absence of connections from the n neurons in the network, and s bits to encode the presence or absence of connections from s sensory receptors (for example, the total string length for a network of 10 neurons and 16 sensory receptors is $(10 \times (1 + 10 + 16)) = 270$ bits). Synaptic weight values are predefined and equal for all expressed weights, and are not evolved. Evolutionary and spiking models are adapted to the specific hardware constraints of the robotic setups and are therefore detailed in each section.

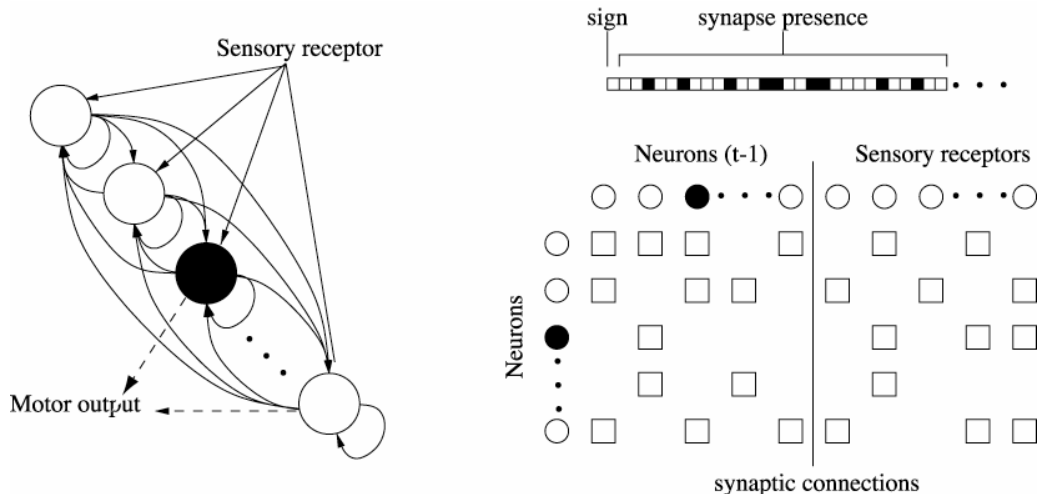


Figure 1. Architecture of the spiking network used in the experiments (only a few neurons and connections are shown) and genetic representation of one neuron (top right). Each neuron can be connected to a number of spiking sensory receptors. Four neurons are used to set the speeds of the two wheels in push-pull mode (the activation of the push neuron determines the forward rotation speed; the activation of the pull neuron determines the backward rotation speed; the algebraic difference between the two speeds gives the resulting rotation speed and direction). *Left.* A conventional representation showing the network architecture (after [1]). *Right.* The same network unfolded in time (neurons as circles, synaptic connections as squares). The neurons on the leftmost column receive signals from connected neurons and receptors shown on the top row. The left part of the top row shows the same neurons at the previous time step, the right part shows the sensory receptors (sensory receptors are not connected to each other). *Top:* A binary genetic string is composed of a number of blocks (one per neuron), each block encoding the sign of the neuron (white = excitatory, black = inhibitory) and the presence of incoming connections from other neurons (white = present, black = absent). Synaptic weights of expressed connections are all equal and not evolved. More detailed information can be found in [5].

So far, the project has been articulated in four stages. In the first stage we evaluated the evolvability of spiking neurons for vision-based navigation in a miniature wheeled robot connected to a desktop computer. In the second stage we simplified the evolutionary and neuron model for implementation in a micro-controller and evaluated the approach on a micro wheeled robot without desktop computer. In the third stage we investigated airborne navigation with an autonomous, vision-based blimp by developing a set of tools to run evolutionary experiments with on-board fitness evaluation and wireless communication with a desktop computer [40]. In the fourth stage, which is still under progress, we developed a set of small indoor airplanes that capitalize upon the technology and algorithms investigated in the third stage.

4. Stage I: Miniature Wheeled Robots

The first stage of the project consisted in assessing the feasibility of evolving networks of spiking neurons for vision-guided navigation [5]. In order to analyze the results, we used the miniature Khepera robot, equipped with a linear CMOS camera, connected to a desktop computer.

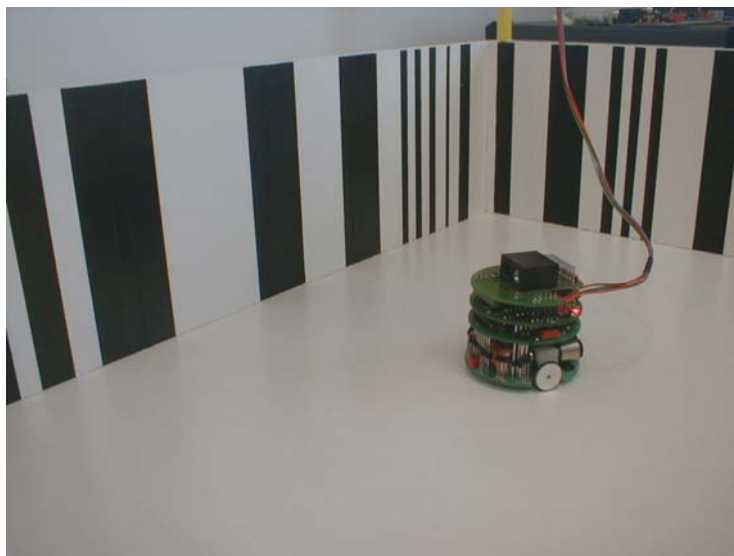


Figure 2. Evolution of vision-based navigation with the Khepera robot. The arena features randomly-arranged contrasted vertical stripes on the walls.

The robot was required to navigate as straight and fast as possible for 40 seconds in a square arena with randomly spaced stripes on the walls. The fitness function was the integral of the wheel velocities in the forward direction. A fully recurrent network of 10 spiking neurons connected to a linear array of 16 photoreceptors of the robot was genetically encoded and evolved on the physical robot. In particular, this experiment was aimed at studying whether functional behaviors can be achieved by simply evolving the connectivity among neurons, but not their synaptic weights (the strength of all expressed connections is set to 1 and the sign is given by the type of pre-synaptic neuron –excitatory or inhibitory).

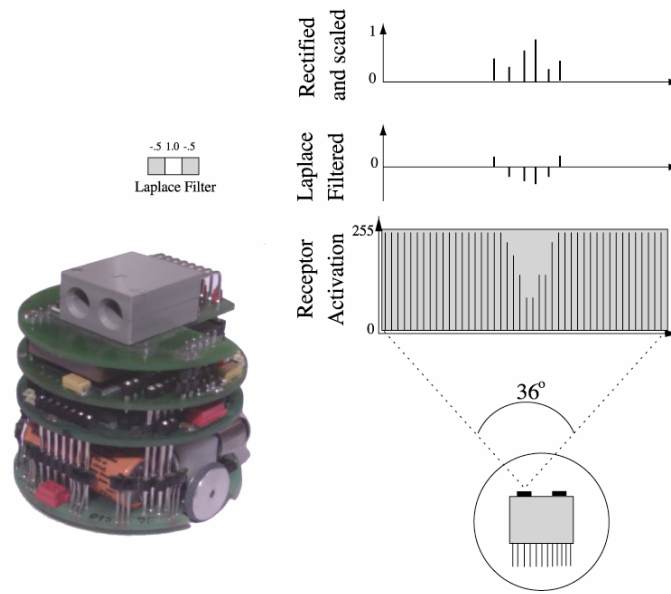


Figure 3. The Khepera robot equipped with a linear vision system composed of 64 photoreceptors. Only 16 photoreceptors are read every 100 ms and filtered through a Laplace filter in order to detect areas of contrast. The filtered values are transformed into positive values and scaled in the range [0,1]. These values represent the probability of emitting a spike for each corresponding neural receptor.

The robot is equipped with a linear CMOS camera of 64 photoreceptors spanning a visual field of 36 degrees (figure 3). In order to reduce the size of the spiking controller and the communication load, we sub-sample the image by reading only one photoreceptor every four. The resulting image of 16 photoreceptors is then convolved with a discrete Laplace filter to detect contrast and the sign is discarded. The resulting values are scaled in the range [0,1] and used as the probability of each corresponding receptor neuron to emit a spike every millisecond.

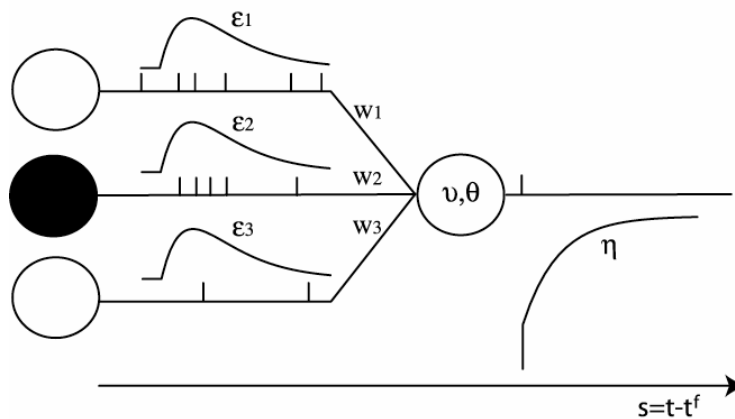


Figure 4. Graphical representation of the Spike Response Model used for the experiments with Khepera and blimp. The variable s is the difference between current time t and the time t' when the neuron fired the last time.

Neuron dynamics are described by the Spike Response Model [12]. The activation v_i^s of the membrane is given by the sum of the synaptic kernels ε_j^s and of the refractory kernel η_i^s (figure 4),

$$v_i^s = \sum_j \varepsilon_j^s + \eta_i^s$$

where i represents the index of the postsynaptic neuron, j the presynaptic one, and s the time elapsed between current time t and the time t_f when a spike was last emitted. The synaptic kernel ε_j^s weighs incoming spikes through an exponential function so that most recent spikes have stronger effect on the activation (or inhibition) of the membrane. The refractory kernel η_i^s models the behavior of the membrane after the emission of a spike. Immediately after a spiking event, the membrane is set to very low negative values and gradually returns to its resting potential. If the combination of incoming weighted spikes and of the membrane potential is larger than the threshold ϑ_i , the neuron emits a spike. The parameters of synaptic kernels, of the refractory kernel, and the value of the thresholds are predefined, equal for all neurons, and are not evolved.

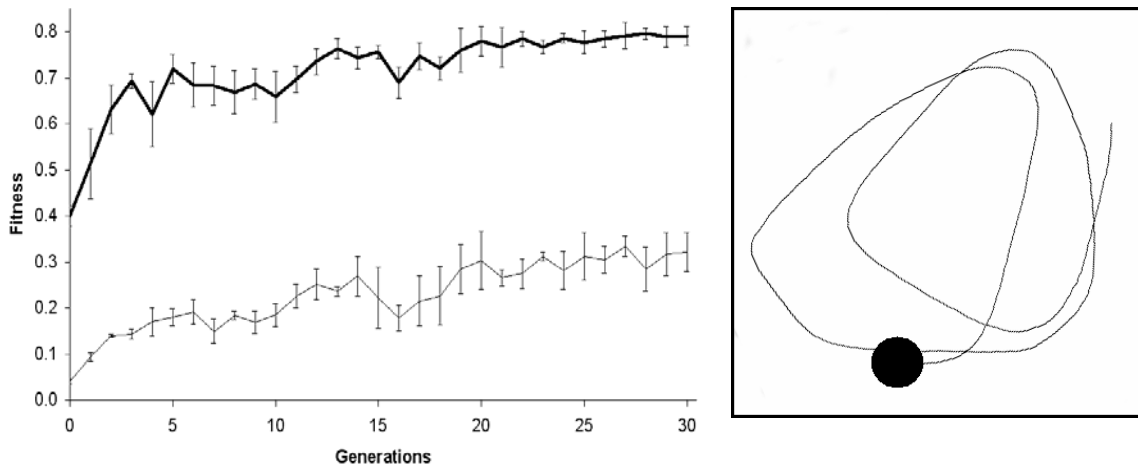


Figure 5. *Left.* Population mean (thin line) and best (thick line) fitness for 30 generations. Data points are averages over three evolutionary runs. A fitness of 0.7 already corresponds to navigation without hitting walls. Fitness 1.0 corresponds to a robot moving straight forward at maximum speed for the entire duration of its life, but it cannot be achieved in this environment where the robot must often turn to avoid walls. *Right.* Trajectory of the best individual of the last generation. Odometric data are obtained from wheel encoders during test.

The neural network and evolutionary algorithm run on a desktop computer accessing sensors and motors of the Khepera robot every 100 ms through a serial connection and rotating contacts. The network is updated every millisecond using the last available sensory data. Less than 10 generations (see figure 5) are sufficient to develop efficient navigation without hitting the walls (corresponding to a fitness of 0.7 in this environment). Further increment in the fitness values correspond to straighter and faster motion. Functional, behavioral, and lesion analysis of evolved networks (albeit in geometrically different environments) are described in [5]. An important

result of that analysis for the scope of this article is that evolved spiking controllers use firing rate instead of firing time. In other words, the activation of neurons are proportional to the number of pre-synaptic spikes within a short time window, but are not determined by the precise time of arrival of single spikes. Therefore, at least in these experimental settings, the potential complex time dynamics offered by the Spike Response Model are not exploited by evolved systems.

5. Stage II: Micro Wheeled Robots

Given the promising results obtained in the first stage, the second stage of the project consisted in developing a low-level implementation of the evolutionary spiking network in a PICTM micro-controller with only 60 bytes of RAM (to store variables), 1k words of ROM (to store program instructions), 60 bytes of electrically erasable memory (EEPROM) to store variables up to 40 years when the micro-controller is not powered, and 4 MHz of clock speed. These types of micro-controllers are suitable for micro-flyers because they require very little power, are extremely small and light, and include most of the circuitry required to interface sensors and actuators.

The EEPROM was used to store the chromosomes and fitness values of the population of evolving individuals while the RAM was used to store all the variables required to update the neuron states and compute the fitness value of a single individual. The ROM was used to store the program instructions necessary to update the network, compute the fitness, read sensors, activate motors, and evolve the connectivity pattern and signs of the neurons.

The evolutionary process was a simplified version of a steady-state tournament-based selection algorithm [39]. An individual chromosome is randomly chosen from the population stored in the EEPROM, copied into the RAM with small random mutations, decoded into the corresponding network and its fitness evaluated for a certain amount of time. If the resulting fitness is larger than the smallest fitness in the population, the mutated chromosome and its fitness is written back into the EEPROM at the place of the individual with the smaller fitness. Otherwise, the mutated chromosome is discarded. Incidentally, the process of reading data from the EEPROM into the RAM is prone to errors that toggle bit values with some small probability [22]. These errors are very rare on a new chip and slowly increase with usage. Therefore, the manufacturer strongly recommends the use of safe read/write routines based on parity check. The reader may notice that these errors could provide a natural source of genetic mutation and thus free memory space otherwise used for safe-reading and for routines of artificial mutations. However, we have not exploited this possibility in these experiments.

The non-linear functions and real-valued variables of the Spike Response Model used in the previous stage could not be implemented on the micro-controllers for reasons of limited space and computational power. However, the results obtained at that stage show that the complexity of that model was not fully exploited by evolved systems. Therefore, in this second stage we used a simpler, discrete-time, integrate-and-fire model with linear leakage and refractory period [7]. The membrane potential is linearly incremented by incoming spikes from excitatory neurons and

linearly decremented by incoming spikes from inhibitory neurons. In addition, at each time step a leakage factor is subtracted from the membrane potential. If the potential reaches the membrane threshold, the neuron emits a spike and for a predefined period it is not sensitive to incoming spikes (refractory period). In the experiments described below, the refractory period is one time step.

This linear model has been implemented using only logic operators available on the micro-controller, such as AND, OR, NOT, and bit shift. The connectivity pattern of a neuron is represented by the individual bit values within a single byte. Similarly, signs and spiking states of each neuron in the network are represented by bit values on a single byte. The activation of a single neuron is computed in parallel by comparing pair-wise the connectivity pattern with the spiking activity of the neurons and with the sign of the neurons. A bit-shift procedure is used to increment and decrement the membrane potential.

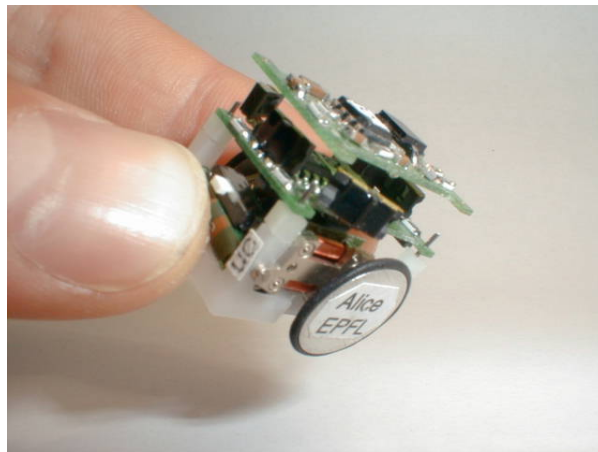


Figure 6. The autonomous micro-robot Alice equipped with micro-controller, infrared active sensors, batteries, and two wheels.

The implementation of a spiking neural network with 8 neurons and 8 sensory receptors on the 8-bit PIC16F628, of its genetic encoding and fitness computation, and of the steady-state evolutionary algorithm took less than 35 bytes of RAM, approximately 500 words of assembly-code, and achieved an update rate of 2 ms for the entire network, which is comparable to the update speed of biological neurons.

The system was then evaluated on the Alice micro-robot (figure 6), which is equipped with the same family of PIC micro-controllers described above, to evolve a navigation and obstacle-avoidance behavior using the same fitness function described in [6].

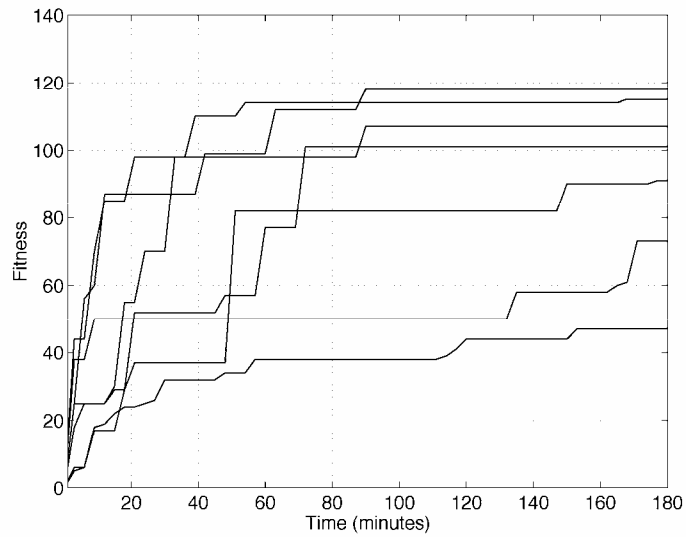


Figure 7. Fitness of the best individuals generated over time. Each line corresponds to a different evolutionary run. A fitness of 100 is sufficient for accurate wall following.

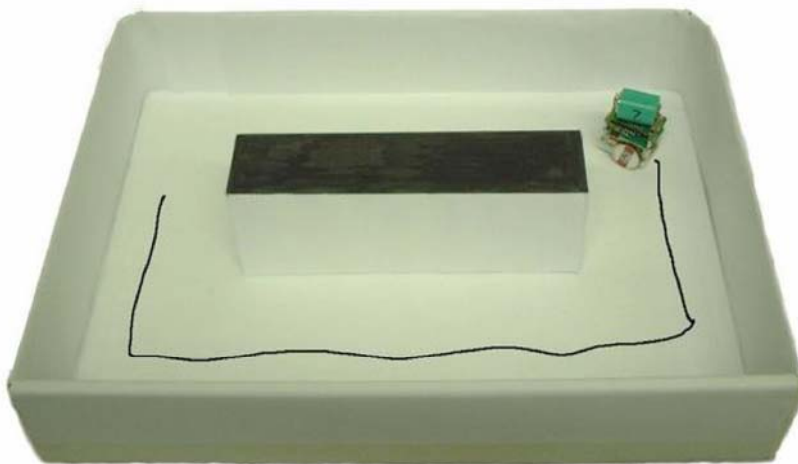


Figure 8. Path covered in 10 seconds by an evolved robot with fitness 120 (see figure 7 above).

It took less than 20 minutes for the robot and its embedded evolutionary algorithm to develop and retain smooth navigation abilities in a simple maze (figures 7 and 8). However, in these experiments we used active infrared sensors, instead of vision, because at that time a vision module was not yet available for the Alice micro-robot.

6. Stage III: Vision-Based Blimp

The third stage of the project consisted in evaluating the evolutionary spiking network for its ability to drive a vision-based blimp in a 5 by 5 meters room (figure 9). In these experiments, we used the same evolutionary algorithm developed in stage I for the Khepera experiments in order to compare the evolvability and results of the two spiking models (the complex one used in stage I and the simpler one used in stage II). Since the two models did not generate significantly different results, the results presented in this section hold for both models.



Figure 9. The evolutionary blimp in its environment, a 5 by 5 meters room with randomly spaced stripes on walls.

The blimp is equipped with two propellers for horizontal displacement and rotation about yaw axis, and one propeller for vertical displacement. It has one active infrared sensor to detect altitude, a linear vision system facing forward, one anemometer to estimate forward speed, Lithium-Polymer rechargeable batteries, a PIC micro-controller, and a Bluetooth™ chip for communication with a desktop computer [40]. At this stage, the entire algorithm is implemented in software running on the desktop computer (<http://asl.epfl.ch/resources/evo/goevo>), which exchanges vision data and motor commands with the blimp every 100 ms. The evolutionary blimp is asked to move forward as fast as possible for one minute using only visual information (during these preliminary experiments, altitude control is provided by an automatic routine based on the vertical distance sensor). The fitness is proportional to the clockwise rotation of the anemometer, which is an estimate of the forward navigation of the blimp. A preliminary set of experiments indicated that artificial evolution generates in about 20 generations spiking controllers that drive the blimp around the room [41]. Robots equipped with evolved controllers did not totally avoid hitting walls since they are not explicitly asked to do so. The fitness function

only encourages high average forward speed but does not penalize contact with obstacles. Evolved robots tend to lean against walls for stopping rotation before accelerating towards the opposite side of the room.

In later experiments (figure 10), we added information from a miniature yaw gyroscope to the neural network input, in addition to vision data. This enables the blimp to display smoother trajectories and almost no collisions with the walls, although the fitness function remained unchanged (video clips are available for download from the project web page: <http://asl.epfl.ch/research/projects/AdaptiveVisionbasedFlyingRobots>).

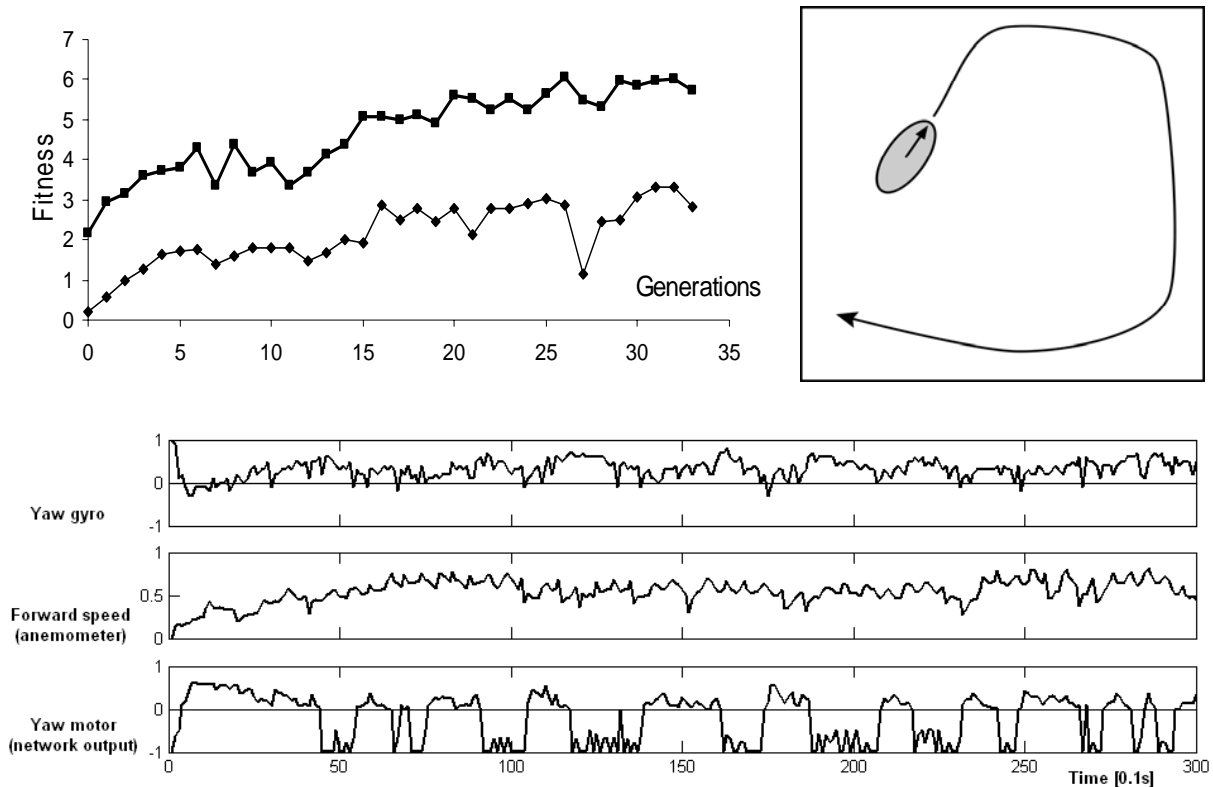


Figure 10. *Top left.* Fitness values of three evolutionary runs (best fitness = squares, average fitness = lozenges). Each data point is the average of three evolutionary runs starting with different random initialization of the chromosomes. *Top right.* Hand-drawn estimation of the typical path (top view) of a best individual in its experimental environment (a 5 by 5 meter room). *Bottom graphs.* Performance of the selected best individual during 30 seconds. The upper graph shows the turning rate (arbitrary unit) about yaw axis. Note that this is only one of the controller inputs. Other vision inputs are not shown. The second graph displays the forward speed (also used as fitness criterion) as measured with the anemometer. The last graph shows the command of the motor responsible for yaw movements as given by the neural network. The forward motor is not displayed since evolution always select individuals that keep maximal forward thrust.

The three bottom graphs of figure 10 display a few informative parameters with there evolution over time that is gathered from a good individual autonomously maneuvering in its arena during 30 seconds. The anemometer values show that after an initial acceleration period, the blimp

reaches a satisfactory forward speed and is able to keep it throughout the testing period. This graph also shows that there were neither frontal contacts with walls, nor backward movements. Although quite noisy, the gyro data suggest the actual trajectory of the blimp with varying curvature radius (as reproduced on top-right graph of figure 10), demonstrating the use of the visual input to control the turning rate. The bottom graph shows the quite complex dynamics of the motor controlling the steering behavior. By comparing the top graph with the bottom graph, one can see that there is no simple correlation between yaw gyro data and yaw motor commands. This is due to the complex dynamics of the blimp involving non-linear aerodynamic damping forces, inertial and added mass effects.

A number of experiments remain to be done with the blimp. These include experiments with different type of cameras or visual preprocessing, and experiments with an additional vertical camera where altitude control is also left to the evolutionary network. These and other experiments are under way at the moment of writing.

7. Stage IV: Indoor Winged Robots

The fourth stage of the project, currently in progress, is the development of a micro airplane capable of autonomous indoor flight. A major requirement of this airplane is to be slow enough to move in a room [25] and to allow on-board vision acquisition, network update, and motor control with the same kind of micro-controller used in stage II of the project.

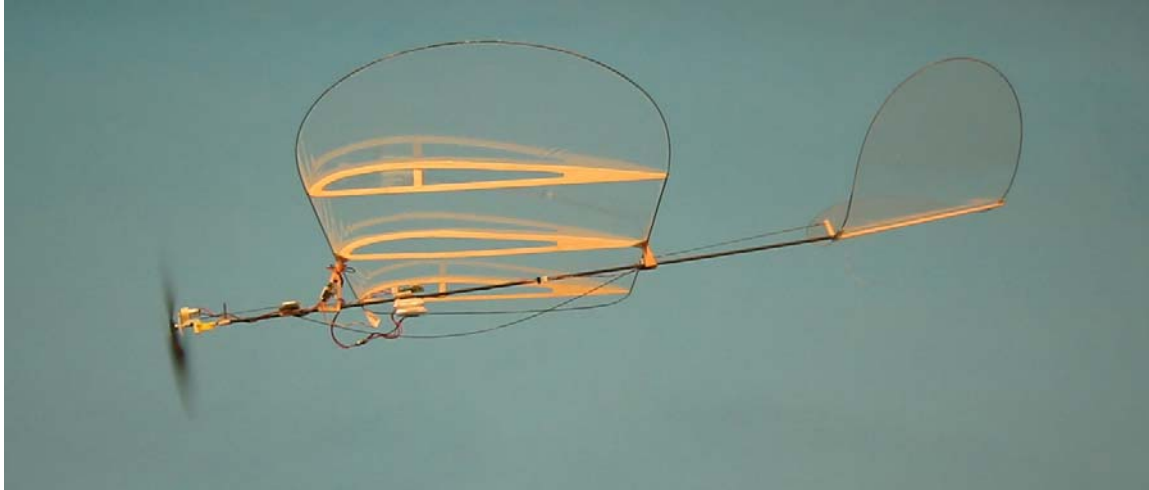


Figure 11. A prototype of the indoor autonomous flyer with bi-directional digital communication using Bluetooth. Main features: weight of 45 g, 82 cm wingspan, 1.4 m/s minimum flight speed, and vectored thrust for steering (the motor-gear-propeller system is articulated about the yaw axis).

Various prototypes have been developed and tested in a wind tunnel. The current prototype, shown in figure 11, weighs 45 grams, has an autonomy of 15 minutes when tele-operated, can fly within a 10 by 10 meters room at walking speed (about 1.4 m/s), and is equipped with batteries, micro-controller, and a BluetoothTM chip for wireless digital communication with a ground-based

workstation. It uses an original solution in order to improve maneuverability at low speed: steering is generated by rotating the thrust system (motor, gear, and propeller) around a vertical axis (vectored thrust). This allows for tight turns within a radius of approximately 2 meters. It has been tested in flight with an additional payload of 10 grams, which is sufficient for a vision system and related microelectronics. A miniature CMOS, color camera has been interfaced to the onboard microcontroller, which is able to grab sub-sampled video stream and send it to the ground station via Bluetooth [40].

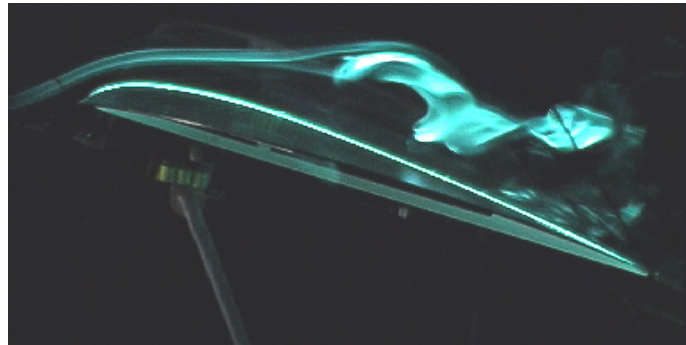


Figure 12. Airflow visualization over the airfoil using a smoke-laser technique within a special wind tunnel at low air velocity. The airplane is attached to the top of a custom-developed device for measuring very small drag and lift forces.

The wind tunnel tests allowed us to optimize the wing structure and airfoil by measuring lift and drag for different models under the same airflow conditions. The measures were obtained with a custom-developed aerodynamic device capable of detecting very low forces. Furthermore, by employing visualization techniques (figure 12), we were able to analyze non-optimal airflow conditions and modify the wing design accordingly.

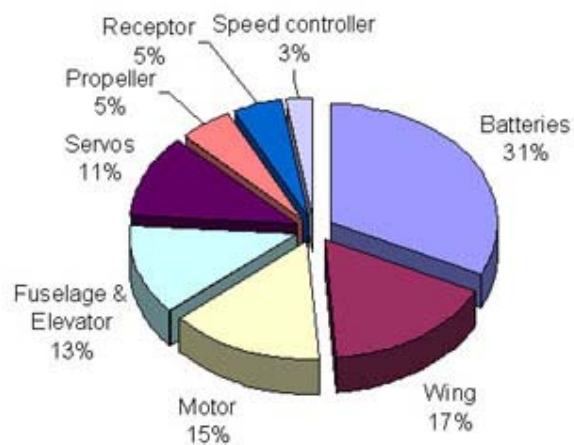


Figure 13. Weight distribution of our prototype indoor slow flyer, the model C4. 100% corresponds to 45 g.

Since aerodynamic lift is proportional to the wing surface and to the square of the relative airspeed, the central issue is to limit the weight while maintaining a wide wing area in order to reach very low flight speed. The building technique is therefore based on carbon fibers and thin plastic film (Mylar™) to ensure good rigidity and low weight.

Figure 13 shows the weight distribution of the aircraft without additional payload (no vision sensor). Most of the mass is in the electronic equipment (batteries, motor, servos, radio, and speed controller). In particular, one third of the weight is in batteries. Therefore, we are currently working with even lighter technologies, such as Lithium-Polymer batteries, magnet-in-a-coil actuators, and pager motors. A fully remote-controlled 10 g airplane is already operational and can fly at only 1 m/s in a 5 by 5 meters sitting room (video clips of this airplane, code-named Celine, are available at www.didel.com).

8. Discussion

The results described in this document represent a first attempt to evolve neural controllers for indoor flying robots. We think that this approach has a number of interesting aspects with respect to hand-coding of precise visuo-motor mechanisms derived from neurophysiological and ethological studies (see section 2.2). An aspect is that evolution can discover alternative solutions to flight navigation exploiting at best the very limited available resources and the specificities of a given artificial body. This was indeed the case of evolved blimp controllers described in [41] where, probably because of the lack of gyroscopic information, the neural controller partly relied on its body to stabilize the trajectory and stop rotations by hitting a wall. Although that strategy is not applicable to a winged robot and was corrected in further experiments by adding a gyroscope sensor (figure 10, top-right), it shows that an evolutionary approach with a simple performance criterion (the fitness is proportional to the amount of forward speed integrated over time) can generate solutions that would not be exploited by a system explicitly designed, e.g., to navigate straight forward and avoid obstacles with saccadic turns whenever characteristic patterns of optical flow will reach a given threshold. Our approach allows the evolutionary process to search for the best behavior adapted to the dynamic and sensory constraints of the robot.

Another interesting aspect is that, once the entire evolutionary methodology has been adapted to the physical constraints of these robots, one can still incorporate pre-wired solutions at the sensory and/or motor interface of the robot and let evolution generate the intervening circuitry. For example, insects with compound eyes use a number of identical sub-circuits repeated all over the eye surface to extract the local component of optical flow (Elementary Motion Detectors). In a recent collaboration with the Neuroinformatics Institute in Zürich, we are exploring this option by integrating tiny neuro-morphic retina chips that incorporate an array of EMDs with analog VLSI technology [30]. The output of this chip, capable of detecting movement direction and intensity, will be directly fed onto the microcontroller running the evolutionary spiking network on the same board. In this case, artificial evolution will have to discover the non-linear mappings between motion information and complex control dynamics of the flying robot. It will be interesting to see whether different image pre-processing techniques affect the mechanisms and

behaviors of evolved systems.

The methodology described here to evolve the spiking circuits for the Khepera, Alice, and blimp robots is not directly applicable to the indoor micro airplane described in the last section above because the airplane cannot autonomously recover from collisions with walls caused by non-functional individuals in the population. To that end, we are developing a simple simulator that captures relevant dynamics of the airplane. The simulator will incorporate a number of parameters measured on the real robot using a custom-made support with micro-force sensors while the robot is in a special wind tunnel at low air velocity (figure 12). Evolution will take place in simulation and the best-evolved controllers will be used to form a small population to be incrementally evolved on the airplane with human assistance in case of imminent collision.

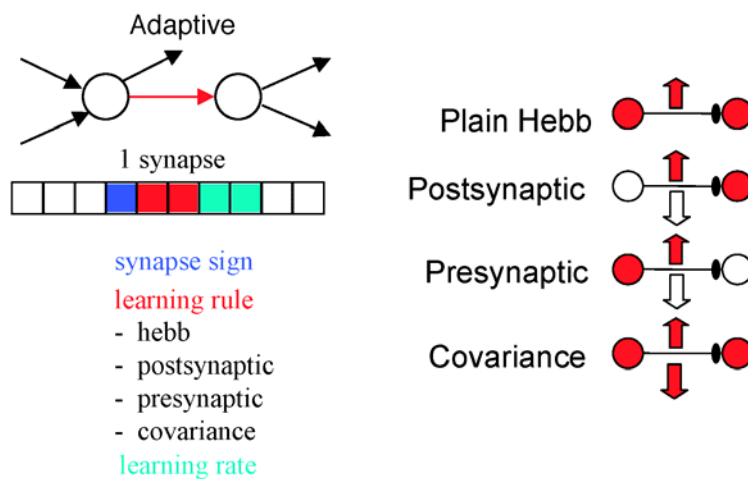


Figure 14. *Left.* Genetic representation of plastic synaptic properties. The chromosome is composed of a set of blocks, one for each synapse in the network. The block for one synapse (in red in the figure) consists of five bits. The first bit encodes the sign of the synapse. The following two bits encode one of four Hebbian learning rules. The remaining two bits encode four learning rate levels. The resulting neural network can use different learning rules and learning rates distributed across different synapses. A variation of this encoding consists of using a single 5-bit block for all the synapses belonging to the same neuron. *Right.* Synaptic modification strengthens (upward arrow) or decrements (downward arrow) synaptic strength according to neuron activity (red = active; white = inactive), but does not modify the sign of the synapse. When a network is created from the genome, all synaptic strengths are always initialized to small random values and can continuously change while the robot interacts with its environment. More details are given in Floreano and Urzelai, 2000.

We anticipate that several evolved neural controllers won't transfer very well because the difference between a simulated flyer and a physical one, or between an airplane in the wind tunnel and one in free flight, is likely to be quite large. This difference may be compensated by fast adaptation of the synaptic connections that map sensory information into motor controls. However, conventional learning algorithms, such as reinforcement learning or other forms of supervised learning, are not applicable because of their long training time, poor convergence properties, and lack of information required by those algorithms (reinforcement signals, correct

motor actions, etc.). The solution that we envisage consists of exploiting artificial evolution to generate mechanisms of fast self-adaptation. Therefore, instead of simply evolving the connectivity of the circuit, as we have done in the experiments described above, we will genetically encode and evolve a set of synaptic plasticity rules and let the neural circuit use them to develop suitable connection strengths from random values literally on the fly (figure 14). This methodology was developed and extensively tested in previous work where we showed that it can generate circuits that adapt very quickly to the environment where they are located [8]. We also showed that such evolved systems transfer very well from simulated to physical robots, and even across different robotic platforms [35].

Our previous work on evolution of plasticity rules was done with conventional neural networks. In that case, the chromosomes encoded four types of plasticity rules, each being a complementary variation of the Hebb rule. These rules will have to be mapped into the temporal domain by taking into account the time difference between pre-synaptic and post-synaptic spikes. Current work on evolution of plasticity rules for spiking neurons, performed within another project, is helping us to explore the best way of implementing such plastic algorithms on micro-controllers.

9. Conclusions

In this paper we have given an overview of our multi-stage, gradual approach, which relies on three key components, namely vision, neuro-morphic circuits, and artificial evolution. Although the project is still in progress, the initial stages reviewed here have already generated a number of results and technologies that are paving the road for the ultimate goal of an autonomous, vision-based, indoor flying robot. The choice of an incremental approach from wheeled to winged robots has been extremely useful for gradual development and thorough tests of algorithms, technologies, and implementations. Along the way, we gradually adapted the neural model and evolutionary algorithms to the electronic constraints of chips that will be used on the flyer, integrated and tested a number of vision modules suitable for horizontal flight, and developed a series of compatible electronic boards ranging from 6 cm to 1 cm in diameter. Although we tried to move to the next stage only once the previous stage was completed, in practice a number of airplane prototypes were already developed right at the beginning of the project in order to provide us with a realistic set of constraints. Also, when moving to the next stage we sometimes realized the necessity to modify some algorithms or technologies and perform comparative tests on robots used in a previous stage. For example, this was the case for the choice and integration of vision modules [40]. The speed of blimps and airplanes required faster and more stable response than that provided by the vision module used on the relatively slower Khepera robot. The newly developed vision boards were then tested in evolutionary experiments on the Khepera robot to check for significantly different behaviors of the resulting neural controllers. Similarly, the compact neural model developed and tested on the micro-robot Alice, is currently being interfaced with a vision module and tested on the Khepera robot before moving on to experiments with blimps and airplanes.

Our choice of evolving spiking neural controllers instead of hand-crafting vision-based control

algorithms on the robotic platforms was motivated by three major reasons. A first reason was given by hardware constraints of limited memory space and computational speed. Here spiking neurons seemed to provide a suitable match to the digital architecture of micro-controllers and evolution seemed to be the only alternative to discover functional wirings of such networks. A second reason was to use evolution in order to explore the space of potential solutions to vision-based navigation without assuming that distance estimation is a necessary step of such systems. A third reason was that bio-mimetic solutions developed so far, such as those described in section 2.2, may not be sufficient to control the complex non-linear dynamics (inertia, aerodynamic effects, turbulence, etc.) of indoor flying robots and may take advantage of an evolutionary approach for architectural and/or parameter tuning.

Whatever methodology and technology will turn out to be most suitable, we expect that adaptability will play a significant role in generating viable control systems and/or adapting them on the fly to a changing and partially unpredictable environment. We are quite optimistic that the combined efforts of our group and of other researchers will result soon in a range of autonomous micro-flying robots.

Acknowledgements. The authors acknowledge significant contributions by Cyril Halter, Michael Bonani and Tancredi Merenda for design of the blimp and airplane, Matthijs van Leeuwen and Antoine Beyeler for experiments with the blimp, Alexis Guanella for modeling and simulation, and Claudio Mattiussi for discussions of spiking neurons. This work is supported by the Swiss National Science Foundation, grant nr. 620-58049.

References

1. Beer, R. & Gallagher, J. (1992) Evolving Dynamical Neural Networks for Adaptive Behavior. *Adaptive Behavior*, **1**, 91-122.
2. Cliff, D. (1991) The computational hoverfly; a study in computational neuroethology. In J.-A. Meyer and S. Wilson (Eds.), *From Animals to Animats. Proceedings of the First International Conference on Simulation of Adaptive Behavior*, Cambridge, MA: MIT Press.
3. Cliff, D., Harvey, I. & Husbands, P. (1993) Explorations in Evolutionary Robotics. *Adaptive Behavior*, **2**, 73-110.
4. Fearing, R.S., Chiang, K.H., Dickinson, M.H., Pick, D.L., Sitti, M. & Yan, J. (2000) Wing transmission for a micromechanical flying insect. *IEEE International Conference on Robotics and Automation*.
5. Floreano, D. & Mattiussi, C. (2001) Evolution of Spiking Neural Controllers for Autonomous Vision-based Robots. In T. Gomi (Ed.), *Evolutionary Robotics IV*, Berlin: Springer.
6. Floreano, D. & Mondada, F. (1994) Automatic Creation of an Autonomous Agent: Genetic Evolution of a Neural Network Driven Robot. In D. Cliff, P. Husbands, J.-A. Meyer, and S. Wilson (Eds.), *From Animals to Animats 3. Proceedings of the Third International Conference on Simulation of Adaptive Behavior*, Cambridge, MA: MIT Press.
7. Floreano, D., Schoeni, N., Caprari, G. & Blynal, J. (2002) Evolutionary Bits'n'Spikes. In R. Standish et al. (eds.) *Proceedings of the International Conference on Artificial Life (ALIFE'02)*, MIT Press, Cambridge, MA.
8. Floreano, D. & Urzelai, J. (2000) Evolutionary Robots with on-line self-organization and behavioral fitness. *Neural Networks*, **13**, 431-443.
9. Franceschini, N., Pichon, J. M. & Blanes, C. (1992) From insect vision to robot vision. *Philosophical Transactions of the Royal Society B*, **337**, 283-294.

10. Franceschini, N., Pichon, J. M. & Blanes, C. (1997) Bionics of Visuo-Motor Control. In T. Gomi (ed.) *Evolutionary Robotics I*, AAI Books, Ontario, Canada.
11. Fujitsu Microelectronics America Inc. (2001) *MB86S02 CIF CMOS Sensor Camera Module*. Preliminary datasheet. <http://www.fma.fujitsu.com>
12. Gerstner, W., van Hemmen, J.L. & Cowan, J.D. (1996) What matters in neuronal locking? *Neural Computation*, 8:1653–1676.
13. Gibson, W. (1979) *The Ecological Approach to Visual Perception*. Houghton Mifflin, Boston.
14. Huber, S., Mallot, H. & Bülthoff, H. (1996) Evolution of the sensorimotor control in an autonomous agent. In P. Maes, M. Mataric, J-A. Mataric, J. Pollack, S. Wilson (eds.) *Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*. MIT Press, Cambridge, MA.
15. Indiveri, G. & Douglas, R. (2000) Robotic Vision: Neuromorphic Vision Sensors. *Science*, **288**, 1189-1190.
16. Indiveri, G. & Verschure, P. (1997) Autonomous vehicle guidance using analog VLSI neuromorphic sensors. In W. Gerstner, A. Germond, M. Hasler & J-D. Nicoud (eds.) *Proceedings of the 7th International Conference on Neural Networks*. Springer Verlag, Berlin.
17. Kroo, I. (1999) The Mesicopter: A Meso-Scale Flight Vehicle. <http://aero.stanford.edu/mesicopter/>
18. Kroo, I., Prinz, F., Shantz, M., Kunz, P., Fay, G., Cheng, S., Fabian, T. & Partridge, C. (2000) The Mesicopter: A Miniature Rotorcraft Concept. Phase II Interim Report.
19. Lewis, M. A., Etienne-Cummings, R., Cohen, A. H. & Hartmann, M. (2000) Toward Biomorphic Control Using Custom aVLSI CPG Chips. In *Proceedings of IEEE International Conference on Robotics and Automation*, IEEE Press.
20. Maas, W. & Bishop, C. (1999) *Pulsed Neural Networks*. Cambridge, MA: MIT Press.
21. Mead, C. (1989) *Analog VLSI and Neural Systems*. Addison-Wesley, Reading, MA
22. Microchip Technology Inc. (<http://www.microchip.com>) (1998) *PIC16F8X. 18-pin Flash/EEPROM 8-Bit Microcontrollers*. Technical specifications.
23. Neumann, T.R. & Bülthoff, H.H. (2001) Insect inspired visual control of translatory flight. In J. Keulemen et al. (Eds.), *Proceedings of the 6th European Conference on Artificial Life*, Berlin: Springer.
24. Neumann, T.R., Huber, S.A. & Bülthoff, H.H. (2001) Artificial systems as models in biological cybernetics. *Behavioral and Brain Sciences*, **24**(6), 1071-1072, Cambridge University Press.
25. Nicoud, J-D. & Zufferey, J-C. (2002) Toward Indoor Flying Robots, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots (IROS'02)*.
26. Pornsin-Sirirak, T.N., Tai, Y.C. & Kennon, M. (2001) Microbat: A Palm-Sized Electrically Powered Ornithopter. *In Proceedings of NASA/JPL Workshop on Biomorphing Robotics*.
27. Reichardt, W. (1969) Movement Perception in Insects. *Processing of optical data by organisms and machines*, New York: Academic, pp. 465-493.
28. Rieke, F., Warland, D., van Steveninck, R. & Bialek, W. (1997) *Spikes. Exploring the neural code*. MIT Press, Cambridge, MA.
29. Ruffier, F. & Franceschini, N. (2003) OCTAVE, a bioinspired visuo-motor control system for the guidance of Micro-Air-Vehicles. *Proceedings of SPIE Conference on Bioengineered and Bioinspired Systems*, Vol. 5119-2.
30. Shih-Chii, L., Kramer, J., Indiveri, G., Delbrück, T. & Douglas, R. (2003) *Analog VLSI: Circuits and Principles*. MIT Press, Cambridge, MA.
31. Sobey, P. (1994) Active navigation with a monocular robot. *Biological Cybernetics*, **71**, 433-440.

32. Sobey, P. & Srinivasan, M. (1991) Measurement of optical flow using a generalized gradient scheme. *Journal of the Optical Society of America*, **8**, 1488-1498.
33. Srinivasan, M., Chal, J.S., Weber, K., Venkatesh, S., Zhang, S.W. & Nagle, M.G. (2000) From Living Insects to Autonomous Robots. In T. Gomi (ed.) *Evolutionary Robotics III*, AAI Books, Ontario, Canada.
34. Srinivasan, M., Zhang, S.W., Lehrer, M., and Collett, T.S. (1995) *Honeybee navigation en route to the goal: Visual flight control and odometry*. *Journal of Experimental Biology*, **199**, 237-244.
35. Urzelai, J. & Floreano, D. (2001) Evolution of Adaptive Synapses: Robots with Fast Adaptive Behavior in New Environments. *Evolutionary Computation*, **9**, 495-524.
36. Webb, B. (2001) Can robots make good models of biological behaviour? *Behavioral and Brain Sciences*, **24**(6).
37. Weber, K., Venkatesh, S. & Srinivasan, M. (1997) Insect Inspired Behaviours for the Autonomous Control of Mobile Robots. In M.V. Srinivasan and S. Venkatesh (eds.), *From Living Eyes to Seeing Machines*, Oxford University Press, 226-248.
38. Whiteside, T.C. & Samuel, D.G. (1970) Blur zone. *Nature*, **225**, 94-95.
39. Whitley, D., & Kauth, J. (1988) GENITOR: A different genetic algorithm. *Proceedings of the Rocky Mountain Conference on Artificial Intelligence*, 118-130.
40. Zufferey, J-C., Beyeler, A. and Floreano, D. (2003) Vision-based Navigation from Wheels to Wings. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'2003)*, pp. 2968-2973.
41. Zufferey, J-C., Floreano, D., van Leeuwen, M. & Merenda, T. (2002) Evolving Vision-based Flying Robots. In Bülthoff, Lee, Poggio, Wallraven (Eds.), *Proceedings of the 2nd International Workshop on Biologically Motivated Computer Vision (BMCV)*, Berlin: Springer.