An Incremental Approach to the Evolutionary Design of Autonomous Controllers for Micro-unmanned Aerial Vehicles

Fabio Ruini, ***Angelo Cangelosi***
Centre for Robotics and Neural Systems
University of Plymouth, UK

Overview

In the context of evolutionary computation, we refer to 'incremental evolution' as 'the process of evolving a population on a simple problem and then using the resulting evolved population as a seed to evolve a solution to a related problem of greater complexity' (Barlow, 2004). Incremental evolution is an alternative approach to the traditional 'direct evolution' methodology, where a population of candidate solutions is evolved from the scratch to tackle a specific task directly aiming to the final solution.
In the work presented here we propose the application of an incremental approach for an Evolutionary Robotics (ER) (Nolfi & Floreano, 2000) model, aimed to the development of autonomous controllers for MAVs (Micro-unmanned Aerial Vehicles) (Ruini & Cangelosi, 2009). Incremental evolution can be seen as a way to force the evolutionary algorithm to generate a punctuated equilibrium evolutionary dynamics, thus manually directing the evolution toward the desired goal following a series of incremental steps.

Experimental setups

The task the MAVs are subject to consists of navigation toward a certain target area - within a three-dimensional obstacle-free environment - relying on a mixture of local and global information. The MAVs match the knowledge about the location of the target, with proprioceptive information related to their current position and orientation, identifying in this way the path to be followed for reaching the target area.
Twelve different topologies of feed-forward neural networks have been tested as controllers for the MAVs. The various architectures differ from each other because of the rotations the aircraft can perform (yaw only, yaw + pitch, yaw + pitch + roll), the absence/presence of a layer of hidden units, and the input encoding used (continuous rather than discrete).
Three different scenarios have been elaborated: Scenario *A* consists of the the tracking of a non movable target by an individual MAV; Scenario *B* involves a moving target attempting to escape from the approaching MAV (5 different moving speeds T_s for the target have been tested: respectively the same speed M_s of the MAV, $M_s/2$, $M_s/3$, $M_s/4$, and $M_s/5$); Scenario *C* requires a cooperative/coordinated operation carried out by a team of 4 MAVs that are expected to reach the target with at least 2 aircraft simultaneously. To achieve the desired goal, the neural network controller used in *C* relies on two additional input neurons compared to the *A* and *B* architectures.

Incremental evolution

The incremental evolution approach applied is different whether the controllers are evolved from Scenario *A* to *B*, rather than from Scenario *A* to *C*. Evolving from Scenario *A* to *B* the connection weights and biases of the individuals belonging to the last generation of the best population evolved in *A* are loaded from the computer memory and used as starting point for the new evolutionary process. Things are slightly more complicated moving from Scenario *A* to *C*, because of the different network topologies used in the latter. In this case, the connection weights coming from the two extra input neurons are added to the A controllers loaded from the memory and their values are set to 0 at the beginning of the second stage of the evolutionary process.
For what concerns incremental evolution from A to B the critical variable for the success of the incremental approach seems to be the complexity of the neural architecture used. For simple networks, as controllers 1-6 (feed-forward NNs without hidden layers) are, the effects seems to be limited. More complex architectures benefited much more instead from the incremental process. This phenomena is evident comparing the results scored by architectures 5 and 6 against controllers 11 and 12. The average success rate of controller 5 dropped by 52.33% and 41.23% for $M_s/2$ and $M_s/3$ respectively, while architecture 6 scored -92.74% and -97.35%. Viceversa, controller 11's performance increased by 67.46% and 97.54%; architecture 12 improved as well, scoring +33.45% and +13.15%. The simplest architectures - as 1, and 2, but also 7, and 8 - did not not show any significant difference in the results obtained following the two alternative approaches. Things are more interesting and variegated for the architectures of intermediate complexity, such as 3, 4, and 9. Architectures 4 and 9 improved their performances among all the parameters measured, for both $M_s/2$ and $M_s/3$. For architecture 3 the fitness values scored (both average and maximum) are pretty similar among direct and incremental evolution, but the average and maximum success rates decreased.
Opposite results have been obtained by the incremental evolution from *A* to *C*, i.e. the further evolution of an architecture specialised in basic navigation to perform a cooperative task. In this case, the only controllers that have gained advantage from the second evolutionary process have been the simplest ones: 1, 2, 7, and 8. All the others architectures (with the only exception of controller 6, for which direct evolution did not succeed) have seen their performances dropping consistently, both in terms of average and maximum fitness, as for what concerns the success rate.

INCREMENTAL VS NON-INCREMENTAL EVOLUTION FOR THE B SETUP ($T_s = \frac{M_s}{2}$)

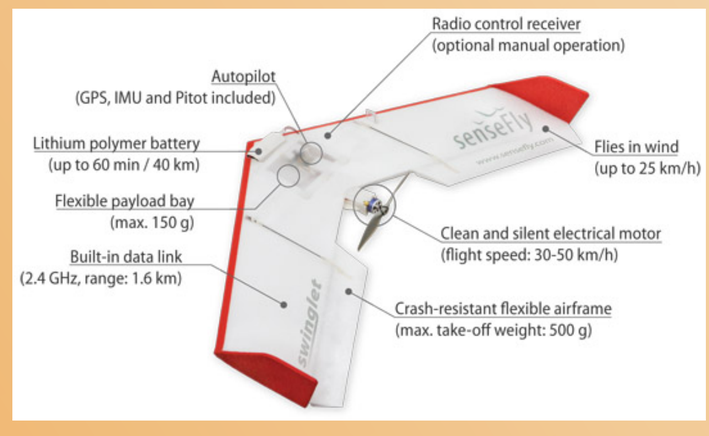
Arch.	Av. fitness	Max fitness	Av. succ. rate	Max succ. rate
1	-2.11%	+0.50%	+1.65%	0.00%
2	+2.62%	+0.99%	+4.82%	0.00%
3	+1.07%	-0.58%	-8.44%	-1.29%
4	+5.29%	+5.51%	+29.15%	+17.93%
5	-6.70%	-4.14%	-52.33%	-16.65%
6	+5.06%	-2.66%	-92.74%	-62.50%
7	+2.40%	+0.96%	+3.97%	0.00%
8	+7.03%	+0.25%	+0.29%	0.00%
9	+16.04%	+5.26%	+25.24%	+3.30%
10	-19.16%	-14.02%	-56.12%	-39.05%
11	-13.80%	+3.59%	+67.46%	+39.10%
12	+10.34%	+11.74%	+33.45%	+43.32%

INCREMENTAL VS NON-INCREMENTAL EVOLUTION FOR THE C SETUP

Arch.	Av. fit.	Max fit.	Av. s. rate	Max s. rate	Half s. rate
1	+2.98%	+1.44%	+4.73%	2.12%	-7.30%
2	+33.20%	+26.53%	+97.76%	+70.21%	-52.22%
3	-2.30%	-5.19%	-26.39%	-18.88%	+15.25%
4	-25.05%	-20.33%	-37.84%	-32.70%	-13.06%
5	-26.37%	-26.90%	-89.12%	-70.25%	-17.36%
6	-6.39%	+12.98%	+1345%	+385.23%	-40.48%
7	+18.69%	+9.11%	+23.96%	+10.26%	+4.51%
8	+12.89%	+4.74%	+21.37%	+8.47%	+1.60%
9	-5.88%	-4.87%	-2.14%	-5.31%	-2.27%
10	-5.62%	-8.25%	-38.22%	-31.75%	+3.89%
11	-28.80%	-28.69%	-76.99%	-65.18%	-25.33%
12	-25.63%	-18.98%	-46.72%	-37.00%	-21.14%

Conclusions

Countless variables seem to have an impact on the profitable applicability of an incremental approach to evolution. The challenge consists in identifying these variables and to provide a theoretical framework - a set of guidelines - that researchers willing to investigate in this area could rely on in the future in order to fully benefit from this approach. Based on the results obtained, we can suggest two critical aspects that must be taken into account. First, it seems to be necessary that the evolutionary algorithm might be free to explore a large space of solutions in order for an incremental approach to be beneficial. Incremental evolution applied from Scenario *A* to *B* has generated significant improvements for the more complex architectures (i.e., those characterised by a larger amount of connection weights), while producing limited benefits for the others. Second, to work properly the incremental process has to go through a series of steps very close to each other, i.e. the incremental complexification of the tasks must follow a smooth path, consisting of minor modifications from one step to the next one This has been demonstrated by the fact that direct evolution, in the testbed scenario, has clearly outperformed the incremental approach for Scenario *C*.
Experiments currently ongoing are aimed to test on a physical robotics platform (namely the senseFly's Swinglet portrayed on the right) the qualitative/'behavioural' differences generated by the controllers evolved according to the different evolutionary methodologies analysed in this work.



Efforts sponsored by the Air Force Office of Scientific Research, Air Office Material Command, USAF under grant number FA8655-07-1-3075.
The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies and endorsements, either expressed or implied, of the Air Force Office of Scientific Research or the U.S. Government.
The U.S. Government is authorized to reproduce and distribute reprints for government purpose notwithstanding any copyright notation thereon.