Coupled Inverted Pendulums: A Benchmark for Evolving Decentral Controllers in Modular Robotics

Heiko Hamann, Thomas Schmickl, Karl Crailsheim
Artificial Life Laboratory of the Department of Zoology, Karl-Franzens University Graz,
Universitätsplatz 2, A-8010 Graz, Austria
heiko.hamann@uni-graz.at

ABSTRACT

The challenging scientific field of self-reconfiguring modular robotics (i.e., decentrally controlled 'super-robots' based on autonomous, interacting robot modules with variable morphologies) calls for novel paradigms of designing robot controllers. One option is the approach of evolutionary robotics. In this approach, the challenge is to achieve high evaluation numbers with the available resources which may even affect the feasibility of this approach. Simulations are usually applied at least in a preliminary stage of research to support controller design. However, even simulations are computationally expensive which gets even more burdensome once comprehensive studies and comparisons between different controller designs and approaches have to be done. Hence, a benchmark with low computational cost is needed that still contains the typical challenges of decentral control, is comparable, and easily manageable. We propose such a benchmark and report an empirical study of its characteristics including the transition from the single-robot setting to the multi-robot setting, typical local optima, and properties of adaptive walks through the fitness landscape.

Categories and Subject Descriptors: I.2.2 Artificial Intelligence: Automatic Programming

General Terms: Theory

1. INTRODUCTION

Typical benchmarks in robotics research are rather practical and playful such as RoboCup [11], DARPA Grand Challenge [21], or see this list [3]. These are of limited applicability to studies in evolutionary robotics [16, 4, 25] which is the study of the synthesis of robot controllers by means of evolutionary computation. Evolutionary robotics demands simple domains [15] which can preferably also be simulated with low computational cost. Studies in evolutionary robotics are differentiated by the exclusive use of simulations [28, 29, 9], by the combined use of simulations and robotic hardware [26], and by the exclusive use of hardware with either a single robot [5, 10] or multiple robots [27].

GECCO'11, July 12–16, 2011, Dublin, Ireland.

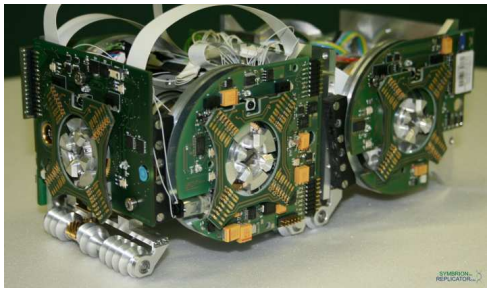


Figure 1: Two connected prototypes of the projects SYMBRION [24] and REPLICATOR [20].

An extremely challenging field within robotics is (multi-)modular robotics [24, 20, 22, 14], which introduces a high degree of flexibility not only in terms of freely programmable agents (robotic modules) but also in terms of freely modifiable body forms (morphologies). In this field, autonomous robotic modules are studied, that are able to physically connect to each other, and can also establish a communication and energy connection (see Fig. 1 for representative prototypes of such modules). Hence, they form a super-robot ('organism'), that is able to re-configure its body shape, see [22, 14] and Fig. 2. Such an organism could be controlled centrally but, as it consists of several autonomous entities already, an obvious approach with a much higher degree of robustness is reached by a decentral (local) control paradigm. In order to reach a maximum of plasticity, each module should have a controller, that is able to generate any necessary behavior, independently of the module's position within the robotic organism.

The variability of body forms of the robotic organism is a strong argument against centralized control and against human-engineered control software, as for every specific (class of) body form different control algorithms or different control parameters need to be designed. In our specific case (see Figs. 1/2), each robotic module has four docking ports, one on each side, to connect physically to other modules. Neighboring modules can dock with 4 orientations (N,E,S,W). Thus, there are $4\times4=16$ configurations to dock two modules. Neglecting the existence of physically unfeasible configurations this gives $\frac{16^{(n-1)}}{4}$ configurations for n modules considering that every configuration has 3 rotation-symmetric equivalents. Although we only consider configurations in

2-d, this gives 4 configurations for 2 modules, 1.72×10^{10} configurations for 10, and 2.08×10^{34} configurations for 30 modules. When considering 3-d body formations, the variability increases even more because modules bending their hinges 'fold' whole organisms into different topologies.
In addition, similarities of organism topologies do not necessarily correspond to similarities in the control algorithms. Small changes in the topology (e.g., rotating one module by 90 degrees) might require very different control techniques. This is all the more relevant once reconfigurations are allowed at runtime.
All these requirements call for a control paradigm of self-adaptivity as it is, for example, common in the field of evolutionary robotics [16, 4, 25]. The robots' modularity and their vast variety of feasible topologies call for control paradigms that allow the exploitation of self-organized processes and combine changes of the topology inherently with a corresponding change in the control software. Possibly after a transient, the controller should again reflect on the body form in a way that allows efficient motion and control of the organism.
Only a few and mostly rather novel controller approaches actually try to address the above requirements (e.g., some evolve artificial neural networks [7], gene regulatory networks [33], or hormone controllers [9]). Whether these approaches can meet these requirements is an open question. Thus, it has to be investigated, if these control software methodologies have enough potential concerning adaptability, flexibility, robustness, self-organization, and evolvability. The evaluations, necessary to select appropriate techniques and to optimize controller designs in combination with one's own concepts, generate a strong demand for fast-evaluating benchmarks which reflect key demands of multi-modular robotics. On one hand, these benchmarks should allow for hundreds of thousands of evaluations within reasonable time, otherwise the evolutionary potential of controller techniques could not be compared statistically. On the other hand, the benchmark should incorporate physical constraints, as the role of physics is fundamentally important in robotics. Multi-modular robotic organisms consist of several physically joined modules, all acting autonomously but within the constraints that are imposed by neighboring modules. This is a crucial aspect which should be reflected in the benchmark used to compare the evolutionary potential of different control paradigms.
To the best of our knowledge, there is currently no well-agreed standard benchmark for modular robotics. Only tests are known that are motivated from the engineering side and that are rather tests for robustness of the hardware itself. Especially concerning evolutionary modular robotics the authors are not aware of any standard benchmark besides the probably too simple task of locomotion [33, 9] which is difficult to compare. Such tasks call for sophisticated simulators [31] with physics engines because the behaviors often rely, for example, on friction. The computational cost of such simulations are high and the speed-up of simulated time vs. real time often drops below 1.
Modular or multi-modular robotics is a rather young scientific field, compared to singular robotics. It is generally characterized by a high degree of flexibility in terms of the manifold forms of interaction between the autonomous robot modules. These modules can operate solitarily, they can interact and cooperate loosely (i.e., for a limited amount of time and without physical connection, cf., for example, the stick-pulling experiment from the related field of swarm robotics [13]), and they can cooperate by connecting physically (e.g., see [9]). Thus, modular robotics incorporates many aspects from classical robotics, swarm or distributed robotics, and in particular aspects from modular robotics itself. We call these different degrees of cooperation the 'coupling' of modules. The coupling between modules is a property that changes continuously (from no coupling in solitary actions to a high degree of coupling in case of physically connected modules) and should be represented in the benchmark.
Also communication is an essential feature of the modular robotics domain. In loosely coupled, swarm-like groups of modular robots the communication methods should be well scalable with increasing module number. In case of tightly coupled modules scalability will usually be a minor issue because high bandwidth communication is available. However, the methods of communication should cope with changes in the topology and also with module breakdowns.

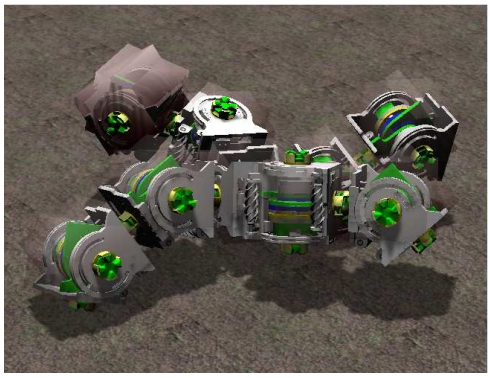


Figure 2: Simulation of the prototypes shown in Fig. 1 using the simulator Symbricator3D [31].

2. DESCRIPTION OF THE COUPLED INVERTED PENDULUMS BENCHMARK

The proposed benchmark is an extension of the well-known inverted-pendulum benchmark (broom balancing). A pendulum, mounted on a cart on a track (i.e., a 1-d world), has to be balanced in the upper equilibrium position. Research of synthesizing controllers for a single inverted pendulum dates back at least to 1964 [30]. Applying evolutionary algorithms to this problem dates back at least to 1990 [12]. During the last 20 years the problem was successfully solved also for even more complex scenarios, such as the double pendulum or the triple pendulum.
Before we explain the extension we applied to the benchmark, we give the equations that we use to simulate each cart with a pendulum, see eqs. 1 through 5, for cart position x, cart velocity v, angular velocity of the pendulum ω, and motor control value u. For the used parameters, see table 1.
We apply several changes to the standard inverted pendulum scenario to increase its complexity and to increase its similarity to challenges of modular robotics scenarios. The most prominent change is that we use multiple carts (or modules) that run all on a single track. The carts are cou-

$$\dot{x} = v \tag{1}$$

$$\dot{v} = \text{motor}(u, v) \tag{2}$$

$$\dot{\phi} = \omega \tag{3}$$

$$\dot{\omega} = \begin{cases} 3g/(2L)\sin(\phi) - 3/(2L)\text{motor}(u,v)\cos(\phi), & \text{if } \omega = 0 \\ 3g/(2L)\sin(\phi) - 3/(2L)\text{motor}(u,v)\cos(\phi) & \\ \qquad -K_p\omega|\omega| - K_l\omega/|\omega|, & \text{else} \end{cases} \tag{4}$$

$$\text{motor}(u,v) = \begin{cases} \begin{cases} \begin{cases} V_{\text{motor}}, & \text{if } v \geq 0 \\ V_{\text{break}}, & \text{else} \end{cases}, & \text{if } u > v \\ \begin{cases} -V_{\text{break}}, & \text{if } v \geq 0 \\ -V_{\text{motor}}, & \text{else} \end{cases}, & \text{else} \end{cases}, & \text{if } |u-v| > \Delta UV \\ \begin{cases} V_{\text{motor}}/\Delta UV(u-v), & \text{if } v \geq 0 \\ V_{\text{break}}/\Delta UV(u-v), & \text{else} \end{cases}, & \text{if } u > v \\ \begin{cases} V_{\text{break}}/\Delta UV(u-v), & \text{if } v \geq 0 \\ V_{\text{motor}}/\Delta UV(u-v), & \text{else} \end{cases}, & \text{else} \end{cases} \end{cases} \tag{5}$$

Table 1: Parameter settings.

parameter	value
gravitational acceleration g	9.81 $\frac{m}{s^2}$
pendulum length L	0.5 m
max. pos. acceleration V_{motor}	7.0 $\frac{m}{s^2}$
max. neg. acceleration V_{break}	8.5 $\frac{m}{s^2}$
world width w	2 m
chain length c	0.35 m
prox. sensor range	1 m
cart width	0.1 m
K_p	0.005
K_l	0.05 $\frac{1}{s^2}$
ΔUV	0.05 $\frac{m}{s}$
initial pos. $x_i(0)$ of cart i	$\mathbf{x}(0) = (-0.4, -0.2, 0)$
initial pos. $\phi_i(0)$ of pendulum i	$\phi(0) = (0.8\pi, 0.9\pi, \pi)$
initial cart velocities $v_i(0)$	$\mathbf{v}(0) = (0, 0, 0)$
initial angular vel. $\omega_i(0)$	$\omega(0) = (0, 0, 0)$
evaluation length $t_{\max}$	4000 time steps

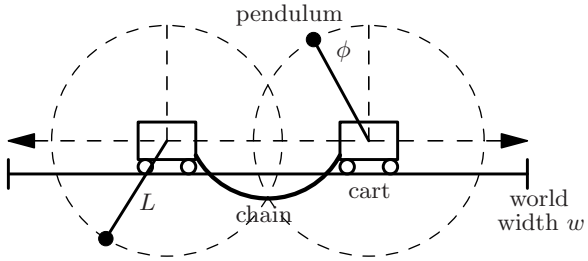


Figure 3: Coupled inverted pendulum benchmark with two carts, pendulums are free to move full 360° mounted on the carts that move in one dimension (left/right) bounded by walls (track ends) and other carts. Marked angle is pendulum angle ϕ.

pled by chains that allow the carts to approach each other closely but the chain length c defines a maximally allowed distance between the carts (see Fig. 3). Carts can move independently as long as they do not pull a chain or run into each other. Hence, each cart has to avoid other carts and walls (cart track ends) and has to balance its pendulum at the same time. Note the difference of this domain from others that mount several pendulums on the same cart, for example, see [32]. In our scenario this would correspond to a chain length of $c = 0$. However, here we are able to define degrees of coupling continuously.

The pendulums are started in lower positions, that is, we include the nonlinear upswinging phase. We also restrict the cart track length resulting in a scenario similar, for example, to that reported in [1]. In combination with the limited acceleration of the cart motor (see V_{motor} in table 1) the upswinging can only be managed by moving back and forth multiple times which increases the complexity of the task.

In addition we limit the sampling rates of all sensors (i.e., a low controller sampling rate). The sampling rates are low which is documented by the relation between the predefined cycle length τ of the controller and the pendulums' maximally allowed angular velocity of $0.05\pi[1/\tau] = 9°[1/\tau]$. Hence, the pendulum can move up to 9° between two calls of the controller and the controller has little time to adapt to new configurations.

In order to adapt the sensor setting to those that are more typical in robotic scenarios, the sensors do not deliver actual angles and positions directly. These values are partitioned onto several sensors and they are also relative rather than absolute (distance to wall instead of the cart's position etc.), see Table 2 for details. This partition is visualized in Fig. 4. For example, sensors S_0, S_1, S_2, and S_3 of the pendulum angle ϕ cover 90° each. All sensor and actuator values have low, discrete resolutions on the interval $[0, 127]$. In addition, the proximity sensors (S_4, S_5) cannot distinguish neighboring robots from walls.

The controllers have two outputs, left actuator A_0 and right actuator A_1 and the acceleration control of the cart is determined by their difference (see Table 2).

In the following experiments we increase the module number (i.e., cart number) without changing the track length. Hence, with increasing module number also the module density is increased which increases the difficulty even more. The modules are controlled locally without global informa-

Table 2: Sensor and actuator setting.

ID	sensor name	syst.-states-to-sens.-values map
S_0	pend. angle 1	$\phi \in [0, 0.5\pi] \rightarrow [127, 0]$, 0 else
S_1	pend. angle 2	$\phi \in [\pi, 1.5\pi] \rightarrow [0, 127]$, 0 else
S_2	pend. angle 3	$\phi \in [0.5\pi, \pi] \rightarrow [127, 0]$, 0 else
S_3	pend. angle 4	$\phi \in [1.5\pi, 2\pi] \rightarrow [0, 127]$, 0 else
S_4	proximity 1	dist. left (max. 1) $\rightarrow [0, 127]$
S_5	proximity 2	dist. right (max. 1) $\rightarrow [0, 127]$
S_6	cart velocity 1	$v \in [-2, 0] \rightarrow [127, 0]$, 0 else
S_7	cart velocity 2	$v \in [0, 2] \rightarrow [0, 127]$, 0 else
S_8	angular vel. 1	$\omega \in [-5\pi, 0] \rightarrow [127, 0]$, 0 else
S_9	angular vel. 2	$\omega \in [0, 5\pi] \rightarrow [0, 127]$, 0 else
A_i	actuators	$A_i \in [0, 127]$, for $i \in \{0, 1\}$
u	motor control	$2(A_0/127 - A_1/127) \rightarrow [-2, 2]$

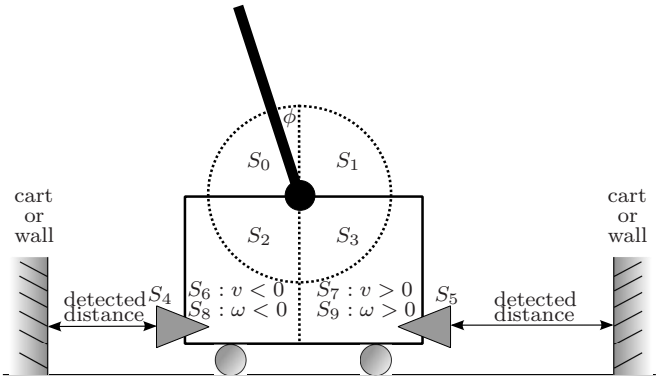


Figure 4: Sketched sensor setting.

tion, ID or positional information and all modules are controlled by identical controllers.

We use an aggregate fitness function [15] which is basically the percentage of time steps that all pendulums spent in the upper equilibrium position ($\phi = 0$):

$$F = \sum_{t=0}^{t_{\max}} \sum_{j=0}^{M-1} \frac{|\phi_j(t) - \pi|}{t_{\max} M \pi}, \tag{6}$$

whereas M is the number of modules and $t_{\max}$ is how long an evaluation lasts in time steps (a typical value is $t_{\max} = 4000$). Deviations from $\phi = 0$ are linearly scaled, that is, $\phi = 0.5\pi$, for example, is evaluated as '50% in upper position'. A fitness of 1 means all pendulums spent all time in the upper position, 0.5 can be interpreted as 'the pendulums spent half the time in upper position', and a fitness of 0 means all pendulums spent all time in the lower equilibrium. An evaluation run is aborted and the fitness is reduced proportionally to the elapsed time once one of the following constraints is violated: cart runs into other cart, cart fully tightens chain, cart runs into wall, pendulum angular velocity too high ($|\omega| > 5\pi \frac{1}{s}$), or cart velocity too high ($|v| > 2\frac{m}{s}$).

The implementation of the cart-pole dynamics is calculated by the Runge-Kutta method of third-order [17] with a discrete time step of size $\Delta t = 0.01$ and is provided online[1].

Search algorithms operating on fitness landscapes related to this benchmark seem to be prone to local optima. Early in an evolutionary run, fast motion of the carts earns good fitness improvements. Subsequently further improvements in the fitness can be reached by spinning the pendulums fast. In a third fitness increase the controller might manage to slow down the pendulums' speed when approaching $\phi = 0$ but the pendulums still spin. Finally, in the absence of noise an evolved controller could generate deterministic cart trajectories that end up with all pendulums at $\phi = 0$. This can be viewed as a local optimum before a fully reactive controller is evolved that actually controls the pendulums also in noisy conditions.

Notice also that conflicting interpretations of sensor inputs exist. A close-by, neighboring cart might be a safe condition in case it moves in the same direction. In contrast a close-by wall might be a dangerous condition.

The proposed benchmark can be extended in many ways. The difficulty can be increased in an uncomplicated way, for example, by upgrading the pendulums to double or triple pendulums. Other useful extensions could be noise models for sensors and actuators, adding static or moving obstacles, or adding a second dimension (i.e., carts move in a plane instead of on a track which introduces new possibilities of collision avoidance). The difficulty could be decreased by more sophisticated sensors, for example, sensors that allow for a distinction between other robots or walls.

3. EMPIRICAL STUDY

In the following we report an empirical study of the benchmark based on our investigations using the Artificial Homeostatic Hormone Systems (AHHS, e.g., see [9, 8]). We report the impact of varied module numbers and couplings, typical local optima, and a fitness landscape analysis.

3.1 Transition from single cart to multiple carts

This benchmark is motivated by the special requirements of decentral control of multiple, interacting modules. The transition from a single- to a multi-module setting is accompanied by a remarkable transition in the performance of different controlling approaches [8]. This is definitely shown in Fig. 5 where best evolved controllers based on the AHHS approach [8] and NEAT [23] are compared (for details see [8]). NEAT's performance for the single-module setting is close to perfect and significantly better than AHHS, whereas AHHS significantly outperforms NEAT in the multi-module setting. This finding justifies the proposed benchmark and calls also for in-depth comparisons of control approaches in modular robotics.

3.2 Varying the module coupling

As discussed in the introduction the coupling of the modules is an important feature of the benchmark. Weak coupling causes the modules to be almost independent of each other, almost comparable to a single robot setting. Medium coupling corresponds in an abstract way to a swarm robotic setting. Strong coupling corresponds to a modular robotics setting because it models the physical connection without forcing neighboring cars into full synchrony. The coupling in the benchmark can be varied by the chain length c continuously from very strong coupling ($c \approx 0$) to weak coupling ($c \approx w$, w is world width).

In Fig. 6 the results of a study based on the AHHS approach [9, 8] is shown (2 modules, linear proportional selection, 200 generations, population size 100). The fitness of the best evolved controller per run for $n = 30$ runs is

[1] `http://heikohamann.de/coupledInvertedPendulums/`

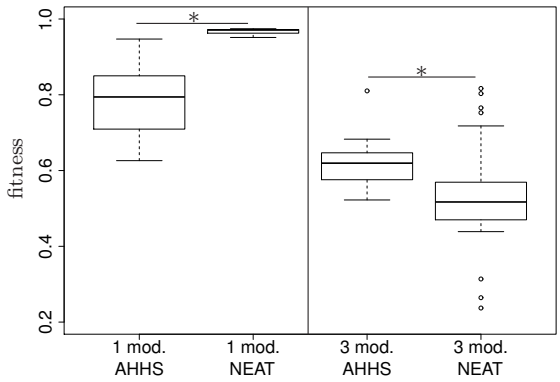


Figure 5: Box-and-whisker plots of the fitnesses of best evolved controllers for AHHS and NEAT approach with 1 and 3 modules, $n = 30$ runs each. Asterisks show significances of $p < 0.05$ using the Wilcoxon rank-sum test; data from [8].

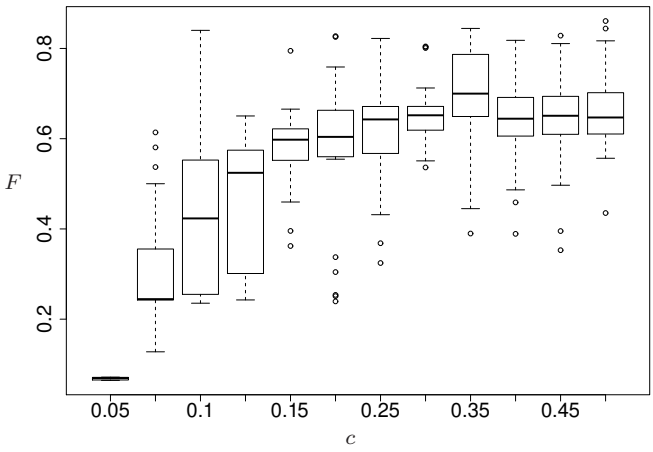
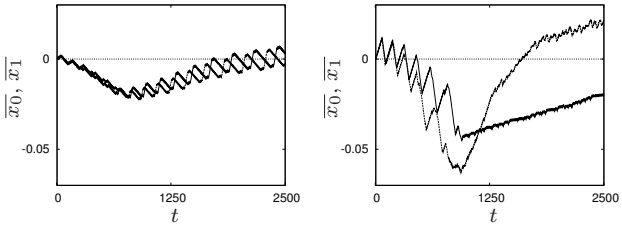


Figure 6: Box-and-whisker plot of the fitness of $n = 30$ evolutionary runs for several chain lengths c (module coupling) using the AHHS approach [9, 8] with 2 modules; significances not shown.

used to compare the complexity of settings with varied chain length c (initial cart positions are adapted accordingly). The task of evolving a successful controller becomes easier with increasing chain length (i.e., decreasing coupling) which is plausible. However, some of the behaviors that evolve for the longer chain lengths would in principle (i.e., scaled to shorter cart distances) also work for shorter chains. It seems that strong coupling prohibits a necessary exploration of the search space in earlier generations of evolutionary runs.

The qualitative change in the motion patterns during the transition from strong to weak couplings is shown in Fig. 7 with the help of two example trajectories. The trajectories are normalized and linearly scaled (see caption) for easier comparison. In case of strong coupling (Fig. 7(a)) the carts move either close to synchrony or asynchronous in form of a simple phase shift. In case of weak coupling (Fig. 7(b)) the carts sometimes move in synchrony as well but sometimes also in independent patterns. The feasibility of this tem-



(a) strong coupling, chain length $c = 0.075$, $F = 0.54$ (b) weak coupling, chain length $c = 0.5$, $F = 0.86$

Figure 7: Trajectories of 2 coupled carts, plotted as superposition with same starting point $\overline{x}_i(0) = 0$ ($i \in \{0, 1\}$) and linearly scaled to $\overline{x}_i(2500) + \overline{x}_i(2500) = 0$ for chain lengths $c \in \{0.075, 0.5\}$ indicating the independent motion in the weak coupling case $c = 0.5$.

porary independence seems to be one cause of the reduced difficulty for bigger chain lengths as shown in Fig. 6.

3.3 Typical local optima

In previous studies of the AHHS approach and NEAT [23] we have learned that the coupled pendulums benchmark (at least in connection with methods of evolutionary computation) is especially prone to local optima. This is, for example, evident in many discrete steps in the best fitness of a single evolutionary run (see Fig. 9). The occurrence of such steps is well known in both artificial and natural systems [2]. However, in this benchmark the local optima are often not overcome which might indicate relatively deep valleys between neighboring local optima.

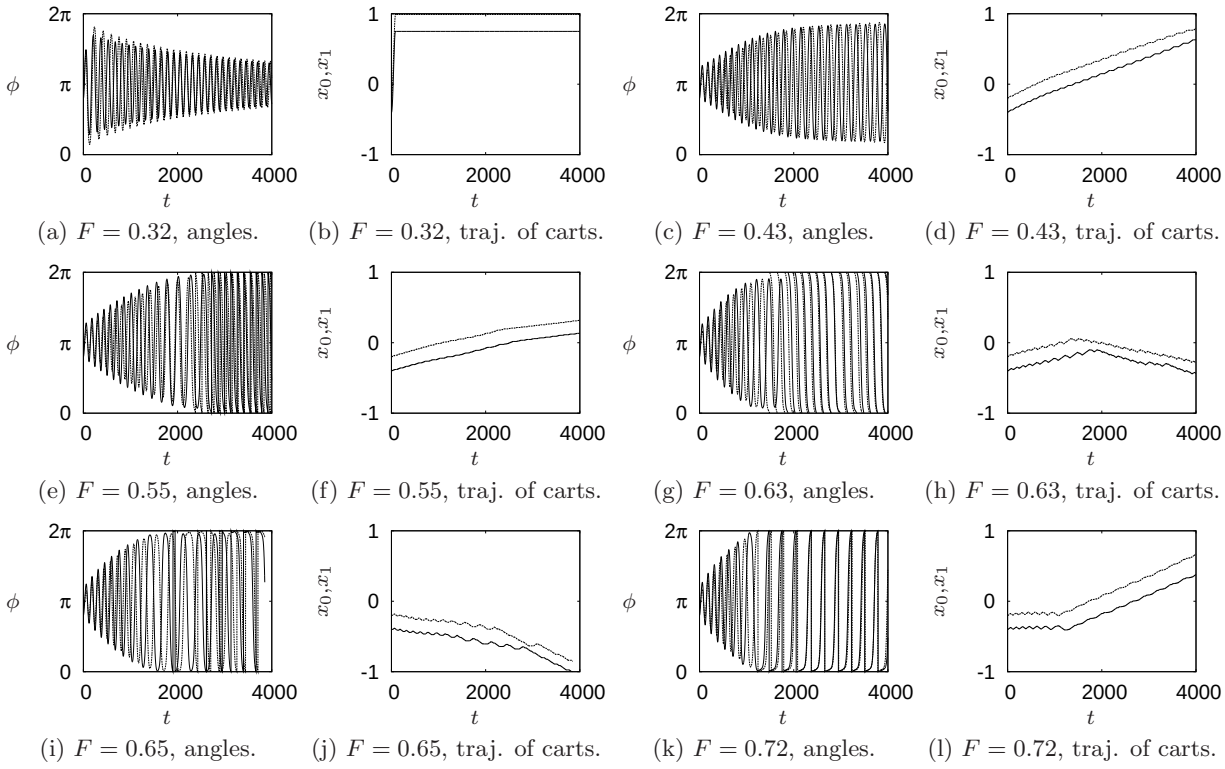
In Fig. 8 we show six typical local optima for a 2-module scenario. Figs. 8(a) and 8(b) show a local optimum because the behavior is based on a fast, unidirectional motion of the carts to the farther bound of the arena which generates only small angular velocities in the pendulums. No additional accelerations are performed after the wall has been approached. An improvement can only be achieved by evolving either an altering stop-and-go motion or bidirectional motion.

Figs. 8(c) and 8(d) show a behavior that generates a unidirectional stop-and-go motion of the carts but results not in an overswinging of the pendulums. Fig. 8(e) and 8(f) show also a unidirectional stop-and-go motion but about at time step 2,300 a first overswinging of the pendulums is observed. Beginning at that time step the stop-and-go motion is slightly changed to keep the pendulums swinging.

Figs. 8(g) and 8(h) show a behavior that achieves the overswinging even before time step 2,000 with a back-and-forth motion which is again adapted to the first occurrence of an overswinging event.

Figs. 8(i) and 8(j) show a qualitatively different behavior because the pendulums are occasionally close to the upper equilibrium position ($\pm 0.05 \times 2\pi$) for up to 160 time steps. In case of the behavior shown in Figs. 8(k) and 8(l) this is achieved in every cycle of the pendulums. The next improvement to this behavior would be something close to the optimal solution, for example, a temporary balancing of both pendulums. The behaviors, that correspond to each fitness plateau, are indicated in Fig. 9.

Whether these local optima are typical for the benchmark and controller–design independent is an open question but



(a) $F = 0.32$, angles. (b) $F = 0.32$, traj. of carts. (c) $F = 0.43$, angles. (d) $F = 0.43$, traj. of carts.

(e) $F = 0.55$, angles. (f) $F = 0.55$, traj. of carts. (g) $F = 0.63$, angles. (h) $F = 0.63$, traj. of carts.

(i) $F = 0.65$, angles. (j) $F = 0.65$, traj. of carts. (k) $F = 0.72$, angles. (l) $F = 0.72$, traj. of carts.

Figure 8: Angles ϕ of the pendulums and cart trajectories x_i for typical local optima, captions give fitness F. Note that keeping the pendulum angles close to $\phi = 0 = 2\pi$ is rewarded with highest fitness.

previous studies indicate this independence at least for two controller designs (AHHS and NEAT, see [8]).

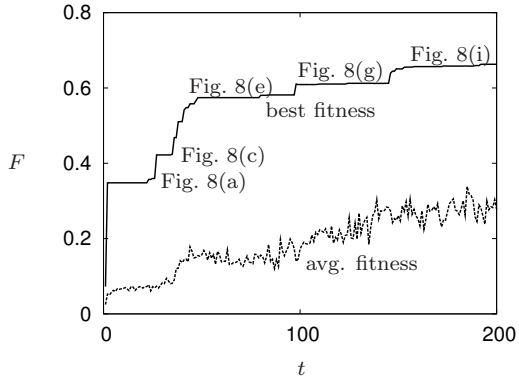


Figure 9: Evolutionary run showing many of the typical local optima (steps in best fitness).

3.4 Fitness landscape analysis

We do adaptive walks through the fitness landscape for a 2-module setting. We start with a random AHHS controller with 5 rules. Here, each rule has 7 relevant and independent features (6 real, 1 discrete variable, sub-rule weights are set to be binary, for details see [8]). This defines the $d = 5 \times 7 = 35$ dimensions of the search space.

The definition of neighbors is not definite in the context of genomes based on floating point values and the evolution strategy [18, 19]. Here we define neighbors by adding and subtracting a defined value $\Delta > 0$. In each step of the adaptive walk we try to find a neighboring controller with higher fitness. Neighboring controllers are reachable from the current controller by changing only one feature by $\Delta > 0$ at a time. This change can be positive (Δ) or negative ($-\Delta$), hence, we get in general $35 \times 2 = 70$ direct neighbors (except for cases when an addition of $\pm\Delta$ would move parameters out of their permitted interval).

In a first analysis, we simplify the benchmark significantly by abandoning all constraints. The carts are allowed to run through each other, the walls and the chains have no effect. The result for $n = 11,128$ adaptive walks is shown in the upper row of Fig. 10. The initial fitness of the random controllers is shown in Fig. 10(a). The main peak at $0.06 < F < 0.07$ with 4002 samples corresponds to an inactive controller. The second peak is at $0.15 < F < 0.16$ with 1762 samples. This fitness corresponds to a controller that is active but moves only erratically.

The fitness of the local optima found at the end of the adaptive walks are shown in Fig. 10(b). The main peak at $0.34 < F < 0.35$ with 1218 samples represents controllers that manage to swing one of the two pendulums fast–a behavior that is difficult to achieve in the standard benchmark due to the chains. In the standard benchmark usually both pendulums or none are swung fast (cf. Fig. 8(e)).

We compare these results to the theoretical results reported in [6] although these are based on Kauffman's NK model. Following [6] the average length of an adaptive walk is defined by $\ln(d)/\ln(2)$, with d is the number of dimensions of the feature vector (genome). For $d = 35$ we get $\ln(35)/\ln(2) \approx 5.129$ which is satisfyingly close to the measured mean length

of 5.803. In accordance with [6] the walk lengths are Poisson distributed except for the outlier of the first bin (walk length 0, i.e., no improvement to the initial random controller is found, see Fig. 10(c)).
In a second analysis, we apply the standard benchmark with all its constraints (see Tab. 1). The result of $n = 12,468$ adaptive walks is shown in the lower row of Fig. 10. In the histogram of the initial fitness (Fig. 10(d)), the main peak at $0 < F < 0.01$ with 7737 samples corresponds to a controller that either runs the carts into each other or into the chains almost instantly (which was not possible in the simplified scenario). The second peak is at $0.06 < F < 0.07$ with 4379 samples. This fitness corresponds to an inactive controller as mentioned above.
The final fitnesses at the end of the adaptive walks are shown in Fig. 10(e). Compared to the initial fitnesses there is only little change. The main peak is now at $0.06 < F < 0.07$ and about 4.6% of the samples reach a fitness above 0.1. It follows that more complex behaviors can hardly be found by pure local search. An additional indicator for a qualitative change of the fitness landscape due to the constraints of the benchmark is the distribution of the walk lengths shown in Fig. 10(f) which is a distribution close to power-law in contrast to the above Poisson distribution (Fig. 10(e)).

4. CONCLUSION

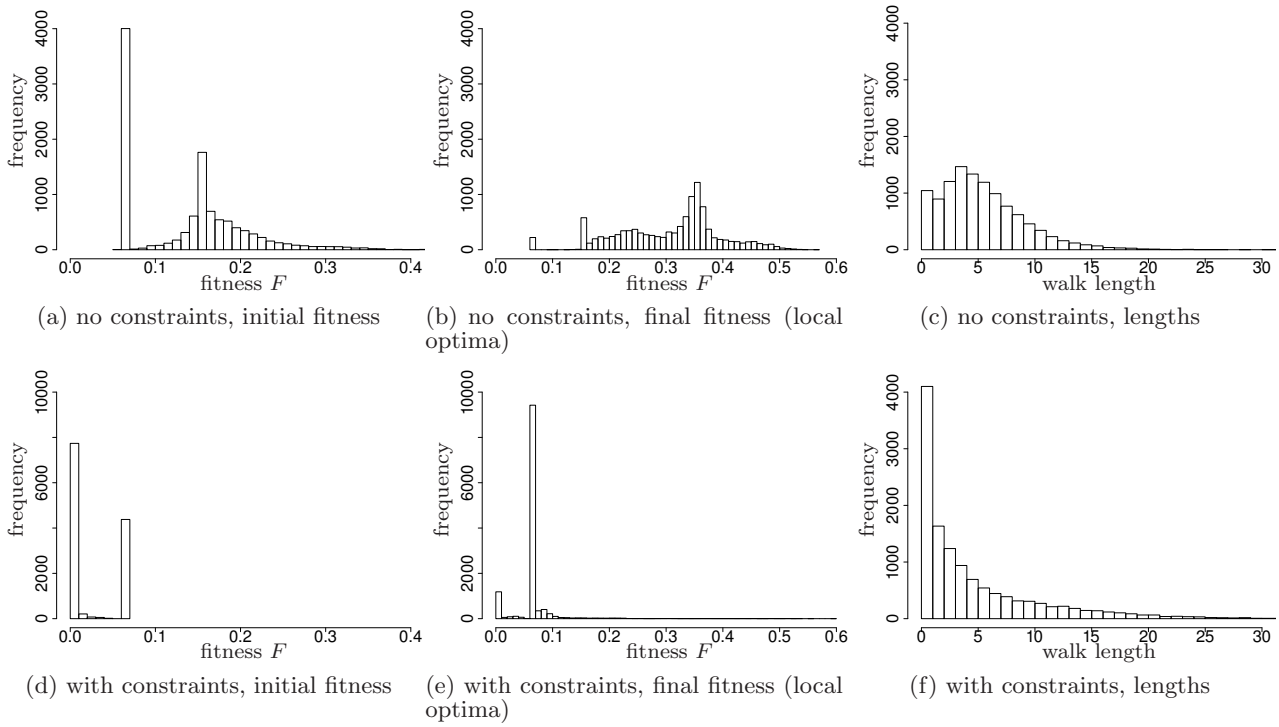
We have presented a novel benchmark for modular robotics with low computational cost that is comparable and easily manageable. It incorporates the typical physical constraints of this field and module coupling is adjustable. The proposed benchmark is a significant extension of the well-known inverted-pendulum benchmark. We introduced multiple carts that are coupled by chains which limit the carts' mobility. Changing the chain length corresponds to a continuous change of the coupling intensities. Weak couplings (long chains) leave the carts almost independent of each other (cf. classical robotics), medium couplings necessitate some interaction between carts (cf. swarm robotics), and strong couplings (short chains) forces the carts to synchronize intermittently (cf. modular robotics).
Another significant addition to the standard benchmark of a single pendulum is that the robots (carts) have to evolve collision avoidance functionality in parallel to other tasks because collisions of carts with other carts or walls (track ends) early in the evaluation are punished by big cuts in the fitness. Thus, in our benchmark with multiple carts, the robots have to evolve three classical tasks in parallel, each task posing significant constraints on the execution of other tasks. First, as in the classical one-cart variant, the task is to balance the pendulum as fast as possible. Second, all carts have to avoid collisions. If one cart moves, this might enforce other carts to move away, an effect which might appear in cascades throughout the group of carts. Third, a wall following behavior ('stay-away-but-not-too-far' task) has to be solved in parallel induced by the chains. The motion of one cart might enforce a neighboring cart to follow.
Empirically we have shown that the benchmark is easier to solve by evolved controllers with increasing chain length as expected. We have reported behaviors that represent typical local optima. In a fitness landscape analysis, we have shown that the fitness landscape changes qualitatively once the constraints of the benchmark are applied (different distributions of adaptive walk lengths likely). In future work we will investigate how manifold control paradigms scale to the decentral multi-module setting as discussed in Sec. 3.1.

Acknowledgment

This work is supported by: EU-IST-FET project 'SYMBRION', no. 216342 and by EU-ICT project 'REPLICATOR', no. 216240.

5. REFERENCES

[1] D. Chatterjee, A. Patra, and H. K. Joglekar. Swing-up and stabilization of a cart-pendulum system under restricted cart track length. *Systems & Control Letters*, 47(4):355–364, 2002.
[2] T. Chouard. Revenge of the hopeful monster. *Nature*, 463:864–867, February 2010.
[3] A. P. del Pôbil, R. Madhavan, and E. Messina, editors. *IROS 2007 Workshop: Benchmarks in Robotics Research*, 2007.
[4] D. Floreano, P. Husbands, and S. Nolfi. Evolutionary robotics. In B. Siciliano and K. Oussama, editors, *Handbook of Robotics*, pages 1423–1452. Springer-Verlag, Berlin, 2008.
[5] D. Floreano and F. Mondada. Genetic evolution of a neural-network driven robot. In D. Cliff, P. Husbands, J.-A. Meyer, and S. Wilson, editors, *From Animals to Animats 3*, pages 421–430, Cambridge, MA, 1994. MIT Press.
[6] H. Flyvbjerg and B. Lautrup. Evolution in a rugged fitness landscape. *Physical Review A*, 46(10):6714–6723, 1992.
[7] E. Haasdijk, A. A. Rusu, and A. Eiben. Hyperneat for locomotion control in modular robots. In G. Tempesti, A. M. Tyrrell, and J. F. Miller, editors, *9th International Conference on Evolvable Systems (ICES 2010)*, pages 169–180. Springer-Verlag, 2010.
[8] H. Hamann, T. Schmickl, and K. Crailsheim. A hormone-based controller for evaluation-minimal evolution in decentrally controlled systems. *Artificial Life*, 2011. submitted.
[9] H. Hamann, J. Stradner, T. Schmickl, and K. Crailsheim. Artificial hormone reaction networks: Towards higher evolvability in evolutionary multi-modular robotics. In H. Fellermann, M. Dörr, M. M. Hanczyc, L. L. Laursen, S. Maurer, D. Merkle, P.-A. Monnard, K. Støy, and S. Rasmussen, editors, *Proc. of the ALife XII Conference*, pages 773–780. MIT Press, 2010.
[10] G. Hornby, S. Takamura, T. Yamamoto, and M. Fujita. Autonomous evolution of dynamic gaits with two quadruped robots. *IEEE Transactions on Robotics*, 21(3):402–410, 2005.
[11] H. Kitano, M. Asada, I. Noda, and H. Matsubara. Robocup: Robot world cup. *Robotics & Automation Magazine, IEEE*, 5(3):30–36, 2002.
[12] J. R. Koza and M. A. Keane. Genetic breeding of non-linear optimal control strategies for broom balancing. In A. Bensoussan and J. Lions, editors, *Analysis and Optimization of Systems*, volume 144 of *LNCS*, pages 47–56. Springer-Verlag, 1990.
[13] A. Martinoli, K. Easton, and W. Agassounon. Modeling swarm robotic systems: A case study in collaborative distributed manipulation. *Int. Journal of Robotics Research*, 23(4):415–436, 2004.
[14] S. Murata, K. Kakomura, and H. Kurokawa. Toward a scalable modular robotic system - navigation, docking, and integration of m-tran. *IEEE Robotics & Automation Magazine*, 14(4):56–63, 2008.
[15] A. L. Nelson, G. J. Barlow, and L. Doitsidis. Fitness functions in evolutionary robotics: A survey and analysis. *Robotics and Autonomous Systems*, 57:345–370, 2009.
[16] S. Nolfi and D. Floreano. *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines.* MIT Press, 2004.



(a) no constraints, initial fitness (b) no constraints, final fitness (local optima) (c) no constraints, lengths

(d) with constraints, initial fitness (e) with constraints, final fitness (local optima) (f) with constraints, lengths

Figure 10: Results of the adaptive walks through the fitness landscape for a simplified 2-module scenario without any constraints and the standard benchmark with the normal constraints (e.g., evaluation is stopped if car runs into wall): distribution of the initial fitnesses of randomly chosen controllers, distribution of the final fitnesses after local optimum is reached, and distribution of the walk lengths ($n = 11,128$ for simplified scenario, $n = 12,468$ for standard benchmark).

[17] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C++*. Cambridge Univ. Press, 2002.

[18] I. Rechenberg. *Evolutionsstrategie. Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Frommann Holzboog, 1973.

[19] I. Rechenberg. *Evolutionsstrategie '94*. Frommann Holzboog, 1994.

[20] REPLICATOR. Project website, 2011. http://www.replicators.eu.

[21] G. Seetharaman, A. Lakhotia, and E. Blasch. Unmanned vehicles come of age: The DARPA grand challenge. *Computer*, 39(12):26–29, 2006.

[22] W.-M. Shen, M. Krivokon, H. Chiu, J. Everist, M. Rubenstein, and J. Venkatesh. Multimode locomotion via SuperBot reconfigurable robots. *Autonomous Robots*, 20(2):165–177, 2006.

[23] K. O. Stanley and R. Miikkulainen. Competitive coevolution through evolutionary complexification. *Journal of Artificial Intelligence Research*, 21(1):63–100, January 2004.

[24] SYMBRION. Project website, 2011. http://www.symbrion.eu.

[25] V. Trianni. *Evolutionary Swarm Robotics - Evolving Self-Organising Behaviours in Groups of Autonomous Robots*, volume 108 of *Studies in Computational Intelligence*. Springer-Verlag, Berlin, Germany, 2008.

[26] P. Vadakkepat, X. Peng, B. Quek, and T. Lee. Evolution of fuzzy behaviors for multi-robotic system. *Robotics and Autonomous Systems*, 55(2):146–161, 2007.

[27] R. Watson, S. Ficici, and J. Pollack. Embodied evolution: Distributing an evolutionary algorithm in a population of robots. *Robotics and Autonomous Systems*, 39(1):1–18, 2002.

[28] S. Whiteson, N. Kohl, R. Miikkulainen, and P. Stone. Evolving keepaway soccer players through task decomposition. In *Genetic and Evolutionary Computation-GECCO 2003*, pages 201–212. Springer-Verlag, 2003.

[29] S. Whiteson and P. Stone. Evolutionary function approximation for reinforcement learning. *Journal of Machine Learning Research*, 7:877–917, 2006.

[30] B. Widrow and F. W. Smith. Pattern recognizing control systems. In J. T. Tou and R. H. Wilcox, editors, *Computer and Information Sciences*, pages 288–317. Clever Hume Press, Washington, DC, 1964.

[31] L. Winkler and H. Wörn. Symbricator3D - A distributed simulation environment for modular robots. In M. Xie, Y. Xiong, C. Xiong, H. Liu, and Z. Hu, editors, *ICIRA*, volume 5928 of *LNCS*, pages 1266–1277. Springer-Verlag, 2009.

[32] X. Xin and M. Kaneda. Analysis of the energy based control for swinging up two pendulums. *IEEE Transactions on Automatic Control*, 50(5):679–684, 2005.

[33] P. Zahadat, D. Christensen, U. P. Schultz, S. Katebi, and K. Støy. Fractal gene regulatory networks for robust locomotion control of modular robots. In S. Doncieux, B. Girard, A. Guillot, J. Hallam, J. Meyer, and J. Mouret, editors, *From Animals to Animats 11*, volume 6226 of *LNAI*, pages 544–554. Springer-Verlag, 2010.