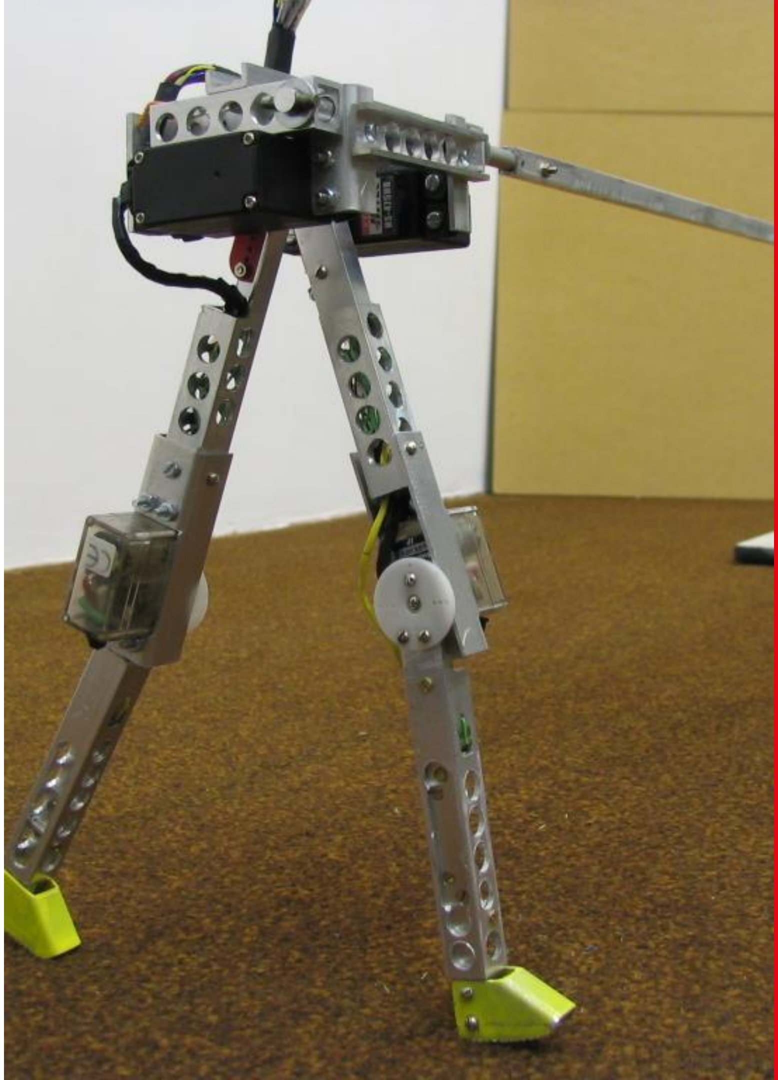
EPFL



Fast biped walking with a sensor-driven neuronal controller and real-time online learning

By Geng et al.

Group 13:

Francesco SAINT-SUPERY

Mathis FINCKH

Legged Robot paper presentation

Table of content

Introduction to RunBot

Executive Summary

- Biped robot, 23 cm high
- 4 actuated joints (hips and knees)
- Simple RC servo-motors for actuation (light and fast)
- Sensors used :
  - RC motors built in potentiometer as angle sensor in each joint
  - Modified piezo transducer to sense ground contact
- Held by a boom
  - Limits to 2D plane
  - Long enough to have little influence on the robot's dynamics. (Falling allowed)
- Curved unactuated feet
  - Light and short helping with fast walking
  - Curved for natural stability
  - Low mass limbs
- Blind robot
- No position or trajectory control

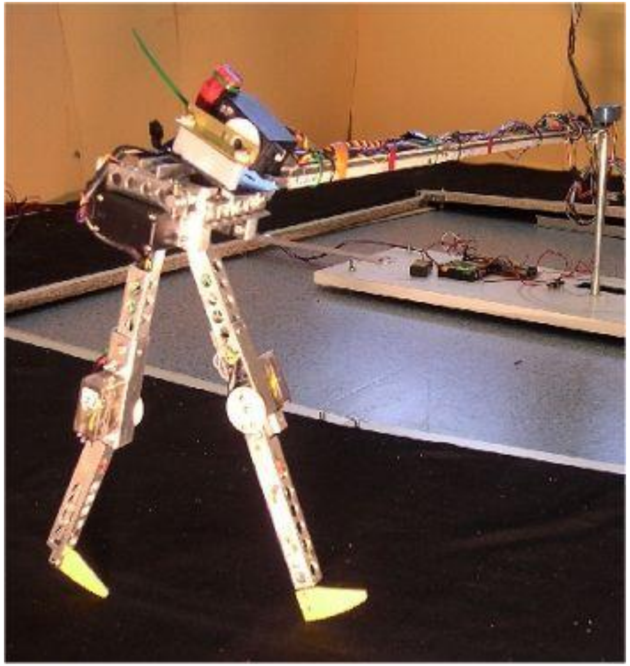


Fig. 2. Walking RunBot attached to the boom

Robot mechanical design

A walking step in 2 stages

- A well-placed center of mass
  - 70% of mass in the trunk
  - CoM placed in front of hip axis to help with momentum issues at low speeds
  - Low mass limbs
- (1) to (2) :
  - Use its own momentum to raise up on the stance leg
  - Covered distance for Com as short as possible
- (2) to (3) :
  - Fall forward naturally
  - Catches itself on the next stance leg

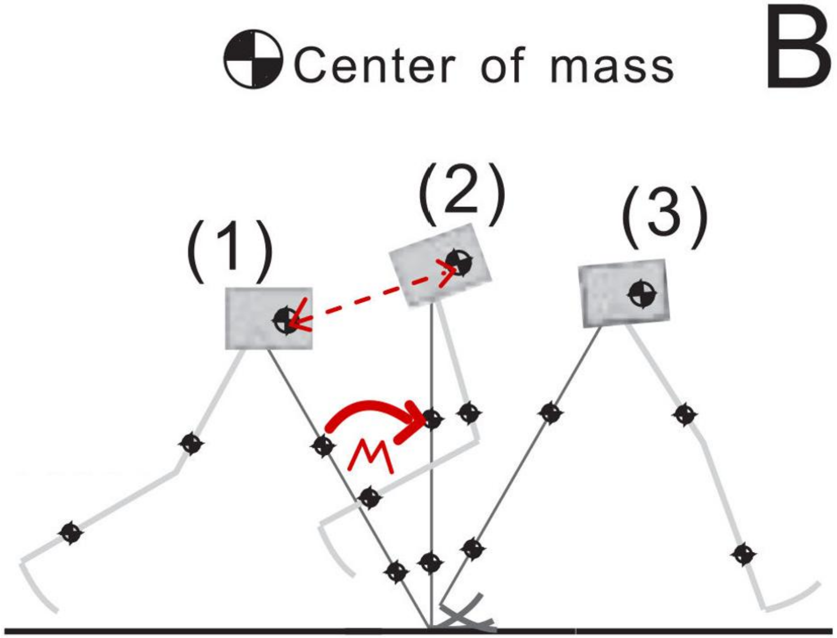


Fig. 3. Illustration of a walking step of RunBot

A Sensor Driven Controller

- **Sensor-driven**
  - Angle sensor (knee and hip)
  - Ground contact sensors
  - Stretch receptors
- Neurons simulated with angle **thresholds** for triggering
- Each sensor has exciting (inhibiting) connections to motor neurons
- Motor **voltage** directly connected to output of motor neurons

Sensor - and motor-neuron models

*See article for more details on models for different neurons

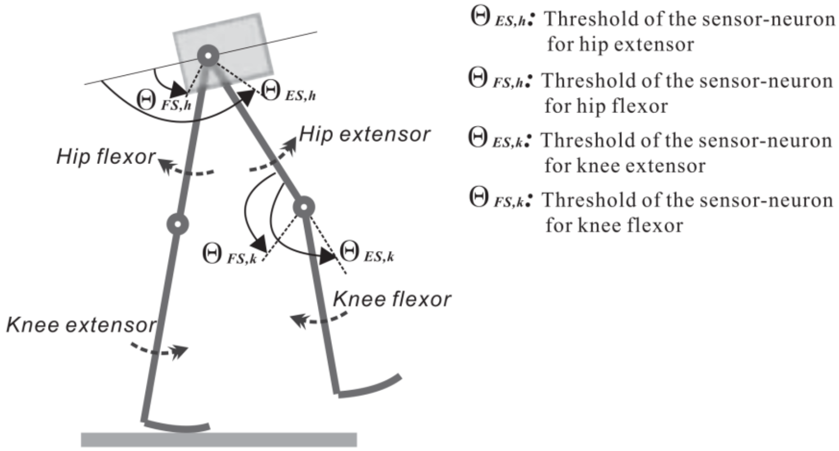


Fig. 4. Control parameter for the joint angle

Hand tuning

- Angle thresholds chosen to mimic normal human gates [annex 1]
- Time constant set to 10 [ms] (normal range of data in biology)
- Synapse weights chosen so that the following importance is kept: angle > stretch > ground contact [annex 1]
  - No big rule for most of these choices
- Neuronal controller advantageous to mode-switching for :
  - smoothness of movement
  - Plasticity (useful for later research)

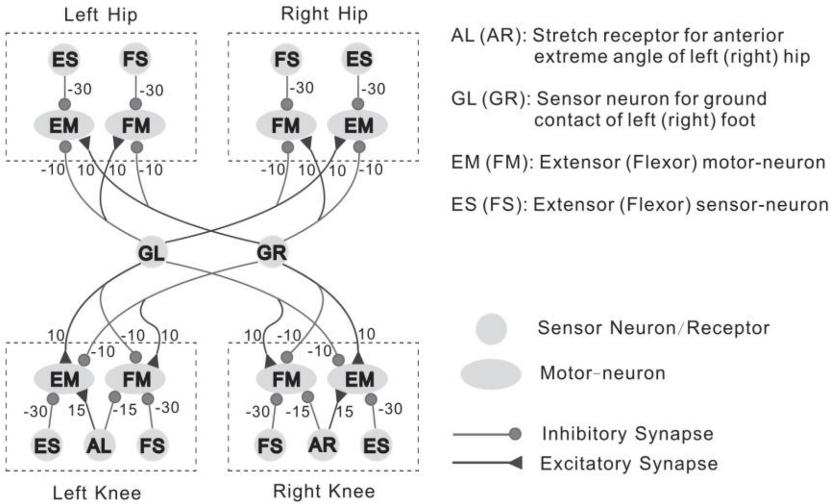


Fig. 5. The neuron model of the sensor-driven controller on RunBot. The small number give the values of the connection weight

Online training/tuning

- No dynamics model, no tracking → no direct mapping of neuronal parameters to walking speed
- But **strong link** between hip motor neuron gain $G_{M,h}$, hip extensor neuron threshold $\Theta_{ES,h}$ and walking speed and gait.
- Formulate as **policy gradient RL problem [annex 2]**
- From random initialization (in stable area), small change in each param and evaluate if increase in walking speed.
- Adjust if outside observed stable area.
- **Can be unstable if initialized in certain areas, even if inside stable area**

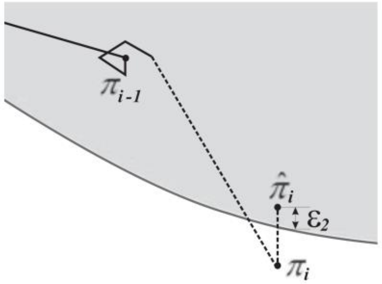


Fig. 7. If the parameter vector π_i is not in the range, it will be pushed in the stable area

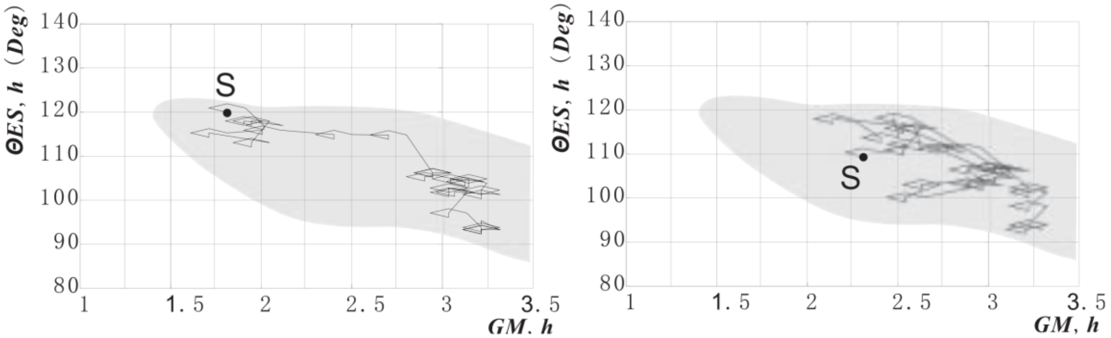


Fig. 6. The shaded areas are the range of the two parameters, in which stable gaits appear. The maximum permitted value of GM,h is 3.45

A Bio – inspired natural dynamic

- Mechanical stop at knee (kneecap), to prevent hyperextension
- Motors voltage = 0 during 1/3 of a step
- Following its natural dynamics dominated by gravity , inertia of the links
- No feedback based active control acts on it
- Only the Sensor-driven controller and the mechanical properties generate the whole gait trajectory
- Similar to animal locomotion → power spike to begin leg swing phase

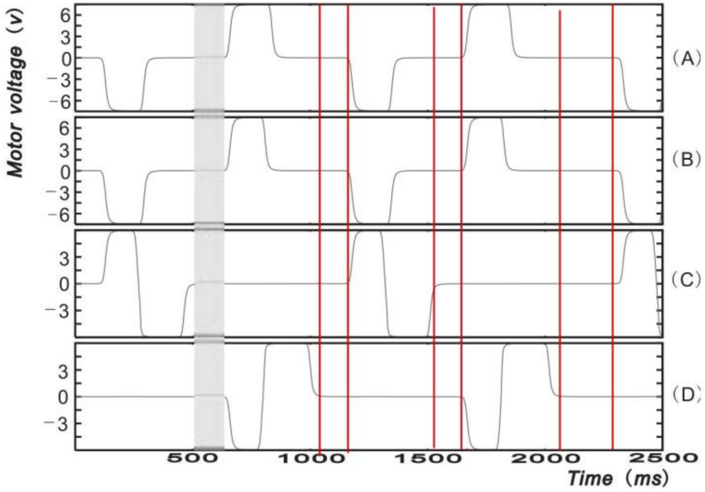
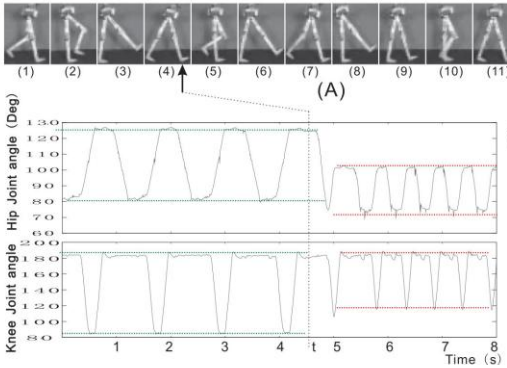


Fig. 8. Motor voltages sent to the servo amplifiers directly from the motor neurons while the robot is walking. (A) left hip; (B) right hip; (C) left knee; (D) right knee. Note that during some period of every gait cycle (gray area), all four motor voltages remain zero and the whole robot moves without actuation.

Results after hand tuning

- The robot achieves wide range of stable dynamic walking gaits
- Robustness of the sensor-driven controller to parameters variations : walking speed changed from slow (0.38 m/s) to fast (0.7 m/s)
- With parameters in central area of figure 10, walking gait show more robustness → tackle obstacles:
  - 9mm low obstacle
  - Walking down a 5° slope



Amplitude reduction with faster gait

Fig. 9. Series of sequential frames of the walking gait. The neuron parameter is changed at the time of frame (4) and time t

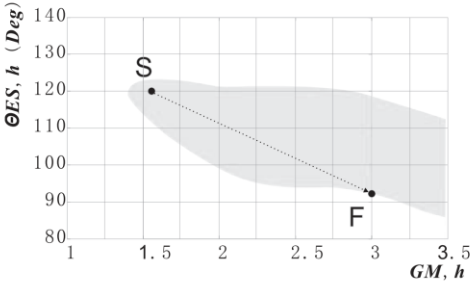


Fig. 10. The shaded areas are the range of the two parameters, in which stable gaits appear.

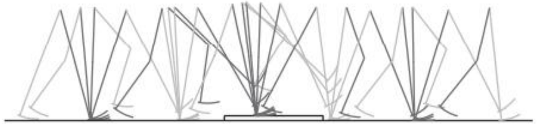


Fig. 12. Stick diagram of RunBot walking over a low obstacle (9mm high, higher ones cannot be tackled).

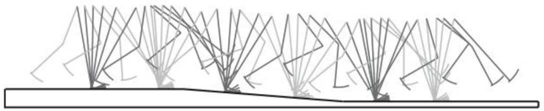


Fig. 11. Stick diagram of RunBot walking down a shallow slope of 5◦ .

Achieving a fast-walking gait with online Policy searching

- Able to reach max speed within 240s of online learning
- Walking speed of about 80cm/s, equivalent to 3.5 leg lengths/s
- Fastest walking robot by its time
- Comparable to human relative walking speed (WR of 4.0 – 4.5 leg lengths/s)
- All of this without any position or trajectory tracking control algorithm
- Detailed online policy **[annex 2]**

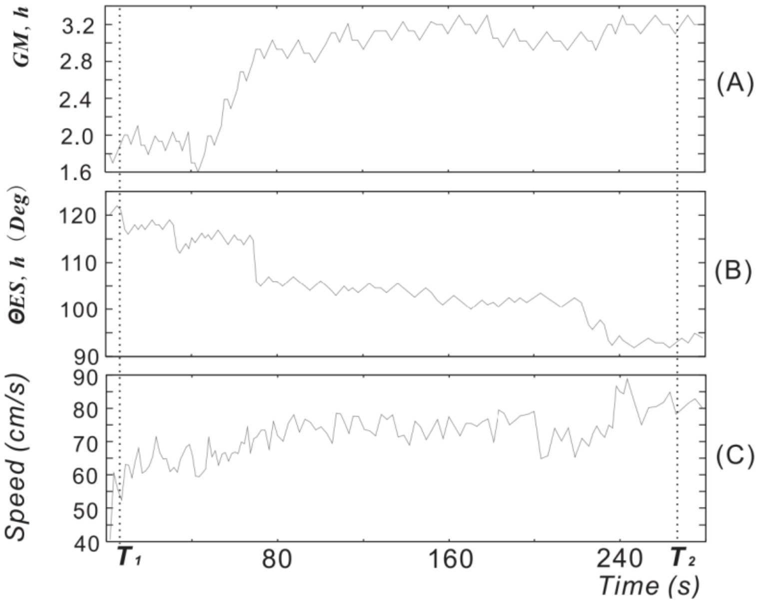


Fig. 12. Real-time data of one experiment. Changes of the controller parameters (A) and (B) and the walking speed (C) during the entire process of learning.

Leg length	0. 8m	0. 75m	0. 9m	0. 67m	0.28m	0. 23m
Max. speed	0. 6m/s	0. 40m/s	1. 25m/s	0. 69m/s	0.25m/s	0. 80m/s
Max relative Speed	0. 75	0. 53	1. 39	1.03	0.89	3.48
Froude Number	0.046	0.021	0.177	0.073	0.023	0.28
	(a)	(b)	(c)	(d)	(e)	(f)

Fig. 15. Relative leg length and maximum relative speed of various biped robots.

Pros and cons

Pros ➕:

- Bio-inspired
  - Intuitive and simple parameter tuning
- No need for a model or much sensing (only angle sensor)
- Fast and efficient thanks to natural dynamics
- No offline training
- Only 2 parameters to tune online

Cons ➖:

- Close to impossible to measure stability
- Hard to predict link between neuronal parameters and walking speed
  - Difficult to debug
- Dependant on hand tuning
- Limited application to real world

Influence, Adoption, and Critique of the Paper

- Cited 200x on Google Scholar
- FWCI: 8.03 (Quite high) (field weighted citation impact)
- Overall remarks :
  - Reflex based controllers are an interesting other option to CPGs
  - Policy gradient methods allow for fewer parameters, but is sensitive to initialisation

- Ijspeert, A. J. (2008). Central pattern generators for locomotion control in animals and robots: a review. *Neural networks*, *21*(4), 642-653.
- Peters, J., & Schaal, S. (2006, October). Policy gradient methods for robotics. In *2006 IEEE/RSJ international conference on intelligent robots and systems* (pp. 2219-2225). IEEE.

Possible exam questions

- <u>What are the 3 key aspects that allowed the RunBot to become the fastest walking robobt of its time ?</u>

Answer :

- Exploiting the natural dynamics of the mechanical structure
- A simple, bio-inspired neuronal sensor-driven controller
- An online policy gradient reinforcement learning algorithm to fine tune speed related parameters

- <u>What make up the bio-inspired part of the RunBot ?</u>

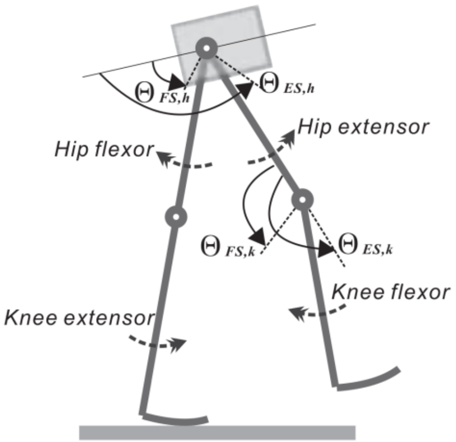
Answer :

- The parameters are tuned to the likness of what is seen in nature (e.g. the thresholds of joint angles, time constant)
- Light legs for speed and reflex like actions, leveraging gravity, allowing 1/3 of the time to be unactuated, which is very efficient
- Sensor-driven control provides smooth motor actuation and mimics the reflex-based control seen in biological organisms

Annex 1

$$W_{GM} \geq \Theta_M + 4$$
$$W_{AM} - W_{GM} \geq \Theta_M + 4$$
$$W_{SM} - W_{AM} - W_{GM} \geq \Theta_M + 4.$$

- $\boldsymbol{W_{GM}}$ **:** Weights of the synapses between the ground contact sensor neurons and the motor neurons.
- $\boldsymbol{W_{AM}}$ **:** Weights of the synapses between the stretch receptors and the motor neurons
- $\boldsymbol{W_{SM}}$ **:** Weights of the synapses between the angle sensor neurons and the motor neurons in the neuron modules of the joints
- angle > stretch > ground contact
- Simply choose : $\Theta_M = 1,\ W_{GM} = 10,$ $W_{AM} = 15, W_{SM} = 30$



$\Theta_{ES,h}$: Threshold of the sensor-neuron for hip extensor

$\Theta_{FS,h}$: Threshold of the sensor-neuron for hip flexor

$\Theta_{ES,k}$: Threshold of the sensor-neuron for knee extensor

$\Theta_{FS,k}$: Threshold of the sensor-neuron for knee flexor

Fig. 3. Control parameter for the joint angle

- Tresholds of the sensor neurons:

$$\begin{cases} \Theta_{FS,k} = 110° \\ \Theta_{ES,k} = 175° \\ \Theta_{FS,h} = 85° \end{cases}$$

Speaker

Annex 2: Policy gradient reinforcement learning

- Initial vector $\pi_0 = (\Theta_1, \Theta_2) = (G_{M,h}, \Theta_{ES,h})$
- Proceeds to evaluate the five polices :
  - $R_1 = (\Theta_1, \theta_2)$
  - $R_2 = (\Theta_1, \theta_2 - \epsilon_2)$
  - $R_3 = (\Theta_1 - \epsilon_1, \theta_2)$
  - $R_4 = (\Theta_1, \theta_2 + \epsilon_2)$
  - $R_5 = (\Theta_1 + \epsilon_1, \theta_2)$
- Construct
- Proceeds to evaluate

- The evaluation of each policy generates a score, Sri , that is a measure of the speed of the gait described by that policy (Ri). We use these scores to construct an adjustment vector A

$$A_1 = 0 \quad \text{if} \quad S_{R1} > S_{R3} \quad \text{and} \quad S_{R1} > S_{R5}$$
$$A_1 = S_{R5} - S_{R3} \quad \text{otherwise.}$$

Similarly,

$$A_2 = 0 \quad \text{if} \quad S_{R1} > S_{R2} \quad \text{and} \quad S_{R1} > S_{R4}$$
$$A_2 = S_{R4} - S_{R2} \quad \text{otherwise.}$$

Then A is normalized and multiplied by an adaptive step-size:

$$\eta = \eta_0(v_{max} - s_{max})/v_{max} \tag{10}$$

Speaker