Just Don't Fall: An AI Agent's Learning Journey Towards Posture Stabilisation

Mohammed Hossny [1,‡,*] and **Julie Iskander** [1,‡]

[1]   Deakin University
[*]   Correspondence: m.hossny@research.deakin.edu.au
[‡]   These authors contributed equally to this work.

Abstract: Learning to maintain postural balance while standing requires a significant, fine coordination effort between the neuromuscular system and the sensory system. It is one of the key contributing factors towards fall prevention, especially in the older population. Using artificial intelligence (AI), we can similarly teach an agent to maintain a standing posture, and thus teach the agent not to fall. In this paper, we investigate the learning progress of an AI agent and how it maintains a stable standing posture through reinforcement learning. We used the Deep Deterministic Policy Gradient method (DDPG) and the OpenSim musculoskeletal simulation environment based on OpenAI Gym. During training, the AI agent learnt three policies. First, it learnt to maintain the Centre-of-Gravity and Zero-Moment-Point in front of the body. Then, it learnt to shift the load of the entire body on one leg while using the other leg for fine tuning the balancing action. Finally, it started to learn the coordination between the two pre-trained policies. This study shows the potentials of using deep reinforcement learning in human movement studies. The learnt AI behaviour also exhibited attempts to achieve an unplanned goal because it correlated with the set goal (e.g. walking in order to prevent falling). The failed attempts to maintain a standing posture is an interesting by-product which can enrich the fall detection and prevention research efforts.

Keywords: Postural Balance; Deep Reinforcement Learning; Postural Stabilisation; Biomechanics

---

1. Introduction

Postural balance is one of the key contributing factors towards fall prevention [1]. Despite the common misconception, maintaining a standing posture does require a huge co-ordination effort between neuromuscular and sensory systems. It also involves coordinated activity of muscles [2, 3]. In contrast, walking was referred to as controlled falling [4]. Walking, which is similar to a swinging pendulum, utilises the potential energy preserved by the upper body and actuated by gravity, where efficiency in walking is maintained by the effective interchange between potential and kinetic energy [4–6]. In addition, from a standing posture, gait is initiated by the de-innervation of the muscles responsible of maintaining balance which causes the body to fall forwards, then a series of coordinated protective steps are performed [7,8]. In postural balance, walking, running and other human movements, the coordinated motor control of the neuro-musculoskeltal units (muscles) plays an essential role [9].

Studies of the biomechanical and physiological aspects of human movement were conducted to understand the neural control needed to maintain the type of movement desired [4,6,10–13]. Most of these studies included human participants recruitment, data collection using motion capture techniques, force plates and/or electromyography electrodes (EMGs). Biomechanics analysis platforms such as OpenSim [14] are also used extensively in different human movement studies [15–22] The main challenge facing this type of studies is the fact that they involve human participants which entails ethical and safety consideration to maintain the safety and well-being of the participants. Simulation based studies using Opensim provided a solution away from human participants [14]. It had been used to predict movement such as walking [23] and jumping [24].

Recently machine learning techniques have been used beside statistical analysis tools in human movement studies with emphasis on classification, prediction and estimation tasks [25]. Bipedal and quadrupedal robotics movement has also been studied extensively where the use of different machine learning techniques were investigated [26–28]. Reinforcement learning (RL) [29] and especially deep reinforcement learning (DRL) that utilised deep learning techniques are slowly but steadily providing solutions to many simulation studies [30–33]. Reinforcement learning is observing an environment and then learning to perform actions that will maximise a score [29]. Learning to maximise the score is done through exploration of the environment and exploitation of past experience, where the challenge is how to create a balance between exploration and exploitation. Continuous action spaces are still imposing a bigger challenge for agents but more sophisticated learners were developed to solve it [34–37]. Recently, DRL was used to train a human musculoskeletal model, based on OpenSim models, to walk and run [31–33] and even to walk with a prosthetic leg [30]. Pilotto et al. has discussed the importance of technology in the advancement of geriatrics and prevention of falls in the elderly population [38]. While the results demonstrated by Lee et al. demonstrate posture stability, their solution presented in [39] relied on providing a reference motion collected from participants. In our work, we focused our efforts on self induced motion leading to posture stabilisation. In this paper, we discuss the use of Artificial Intelligence (AI) and especially DRL to learn how not to fall. More specifically, we investigate the learning progress of an AI agent towards maintaining a standing posture. We adopted a modular design of the control neural network by separating the observation from the policy. We also used multiple policies, each trained separately, as well as a coordination policy to coordinate between the learnt policies. The rest of this paper is organised as follows. Section 2 describes the materials and methods used in this study. Section 3 reports the results. Section 4 discusses the behaviour learnt by the AI agent. Finally, Section 5 derives conclusions and introduces to future directions.

2. Material and Methods

In this work, we focus our efforts on studying the muscle control strategies an AI agent can learn in order to prevent falling. This is achieved via biomechanical simulation using OpenSim and reinforcement learning via deep neural networks. The biomechanics simulation serves as the environment and the deep neural network serves as the motor control part of the brain.

2.1. Biomechanic Simulation Environment

The used environment is based on OpenSim [31], and OpenAI gym [40]. The human musculoskeletal model used is based on the previous works presented in [41–43], where the model is made up of seven body parts. The head, neck, torso and pelvis are represented as a single body for simplicity, in addition, each leg is represented by three body parts, upper leg, lower leg and foot. The model has 14 degrees-of-freedom (DoF), 6 DoFs (3 rotational and 3 translational) for the pelvis, 2 rotational DoFs for each hip, 1 rotations DoF foe each knee and finally, 1 rotational DoF for each ankle. The model is actuated by 22 muscles [44], 11 muscles for each leg. The muscles actuating each leg included the hip adductor, hip abductor, hamstrings, biceps femoris, gluteus maximus, iliopsoas, rectus femoris, vastus intermedius, gastrocnemius, soleus, and tibialis anterior. In addition, contact with the ground is modelled using the Hunt-Crossley model [45]. As shown in Fig. 1, for each foot, contact spheres are positioned at the heel and toes. In addition, another rectangular contact plane is placed over the ground. Force is generated when the objects come into contact, which depends on the velocity of the collision and depth of penetration of the contact objects [31].

The observation fed to the AI agent included 100 values covering ground reaction forces, pelvis velocities, joint angles and muscle state. These readings were grouped as listed in Table 1. Three values were providing random numbers, summarising velocity vector field, from the environment were incorporated to introduce a randomisation factor to the AI training. Typically, the AI agent should learn how to progress within the environment by observing the score provided by the environment. This score reflects how good or bad the AI agent behaved towards achieving the desired goal. It is
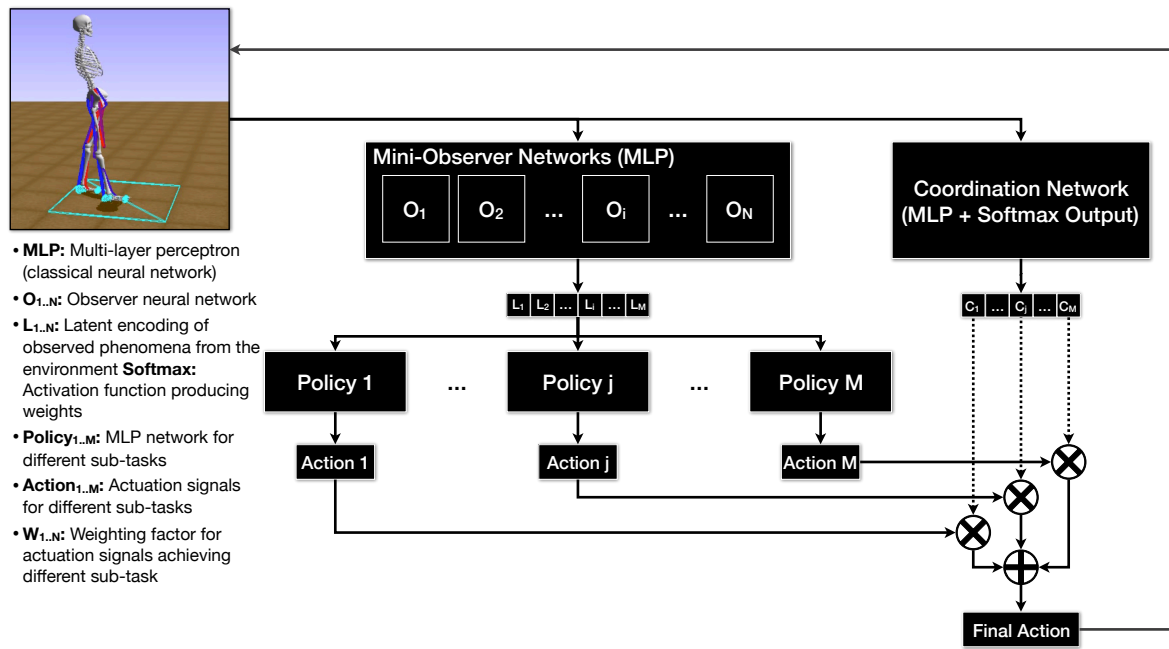
Figure 1. The actor part of the proposed AI agent. The observations (from different modalities) read from the environment, are fed into light weight MLP networks (mini-observers) to produce an encoded latent features. The encoded features are then fed into all policies to produce a set of actions. The produced actions are then multiplied by the weight factors produced by the coordination network based on the state reported by the environment. The final, weighted, action is then fed back into the environment to actuate the muscles.

worth noting that the AI agent must be completely oblivious to what the desired goal is and thus, it should learn based on the praise it receives from the environment (much like a toddler learning to stand and walk). The scoring function was designed to provide a positive reward for having contact with the ground via $\mathbf{F_g}$ while penalising undesired actions resulting in change in pelvis height H, and velocity $||\dot{\mathbf{p}}||, ||\dot{\theta^\mathbf{P}}||$. In order to maintain a standing posture the score was penalised with the magnitude of values and change in the joint angles $||\theta^\mathbf{J}||, ||\Delta\theta^\mathbf{J}||$. The score also includes penalty for the change in joint angles in order to prevent the scissor legs posture described in [30]. This was achieved by penalising the adduction angles $\theta^\mathbf{J}_{add}$ of both legs. The muscular state was not incorporated in the score function. The scoring function is then formulated as

$$R_{GRF} = \begin{cases} 1, & ||\mathbf{F_g}|| > 0 \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

$$R_{Alive} = 1 \tag{2}$$

$$Cost = ||\mathbf{m^a}|| \tag{3}$$

$$+ \quad 8(H - H^s)^2 \tag{4}$$

$$+ \quad 8||\dot{\mathbf{p}}|| + 8||\dot{\theta^\mathbf{P}}|| + 64||\theta^\mathbf{J}|| + 64||\Delta\theta^\mathbf{J}|| \tag{5}$$

$$+ \quad 8\gamma^2 + 8\phi^2 \tag{6}$$

$$+ \quad 512\,\theta^\mathbf{J}_{add} \tag{7}$$

$$Norm.\,Score = R_{Alive} + R_{GRF} - Norm.\,Cost \tag{8}$$

where H^s is the baseline pelvis height when standing, and $||\cdot||$ is the Euclidean norm. All angles were normalised by π. The *Norm. Cost* term is the normalised cost using the sum of the weights.

Table 1. Observation Table

Observation	Size	Notation (in score fn.)	Comments
Ground Reaction Forces	6	$\mathbf{F_g}$	3 per foot
Pelvis Orientation/Linear/Angular Velocity	9	$\dot{\mathbf{p}}, \dot{\theta}^{\mathbf{p}}, H, \gamma, \phi$	
Joint Angles	8	$\theta^{\mathbf{J}}$	4 per leg
Change in Joint Angles	8	$\Delta\theta^{\mathbf{J}}$	4 per leg
Muscle Actuation	22	$\mathbf{m^a}$	11 per leg
Muscle Force	22	$\mathbf{m^f}$	11 per leg
Muscle Length	22	$\mathbf{m^l}$	11 per leg
Random Values	3		velocity vector field [30]

2.2. Artificial Neural Network Model

For any AI agent, the choice of an artificial neural network (ANN) is largely affected by the inputs provided from the environment for several reasons. First, the dynamic range of input values from the environment differs based on the measured phenomenon. For example, joint angles range from $[-\pi, +\pi]$ and increase on a spherical scale while muscle length is normalised between $[0, 1]$ and increases linearly. Even with normalisation of joint angles as $\bar{\theta}_i = \theta_i / \pi$, the rate of change remains different. The second challenge was how information leakage from different measured values during the training of the ANN. To address these two challenges, we designed smaller neural networks to act as mini-observers trained for encoding the input values of the phenomena sharing the same dynamic range and purpose in the simulation. The output of these mini-observers were then concatenated into one encoded output.

The complexity of the task also affected the design of the ANN. Specifically, during the training of the neural network, it attempts to minimise the error between the estimated output and the target output via calculated update gradients. The update gradients update the parameters of the neural network at once which prevents the network from solving sub-tasks (e.g. joint flexion and extension) in order to solve the desired task (e.g. do not fall). In order to address this challenge we chose to train each policy separately with different initialisation parameters and then train a coordination policy to derive a mixture of the actuation signals from different policies. There are two solutions to this problem. The first solution is to enable only certain parts of the neural network to train while locking the rest of the network. The other solution, which is bio-inspired, is to expand the network as needed. In this work, we manually adapted the second solution, by training different policies separately and then integrating them with the coordination policy. Ideally, this approach should be done automatically by using the Neuro-Evolution of Augmenting Topology algorithm (NEAT) [46]. The NEAT algorithm is an evolutionary algorithm which relies on generating several MLP architectures and harnessing the power of mutations and cross-over for exploration of thousands for generations. However, the NEAT algorithm is more suitable to figuring out the MLP topology (i.e. policy). To the best of our knowledge, there is no evidence it could be expanded for an entire architecture of deep neural networks especially with the computationally expensive training and the vast amount of data and/or trials requirement. While UBER is now investing the expansion of the NEAT algorithm to deep learning [47], perhaps the meta-learning research [48] is the closest approach to expanding the NEAT algorithm on an entire deep learning architecture.

To that end, as shown in Fig. 1, the AI model proposed in this work consists of three neural networks; mini-observer networks (one for each modality provided by the environment), policy networks (one for each sub-task) and a coordination network to combine the actions from different policies into a final actuation signal.

2.3. Reinforcement Training Procedure

Due to the dynamic nature of the problem, we chose to utilise deep reinforcement learning. We adopted the Deep Deterministic Policy Gradient (DDPG) method because of its impressive results

in continuous action spaces [34]. The DDPG model consists of an actor network (described above) and a critic network to evaluate the actions produced by the actor in relation to the environment. The critic network takes the actions produced by the actor network and the values obtained from the environment and produces a score [34,49,50]. In our setup, the actor is the entire coordinated multi-policy shown in Fig. 1 while the critic is a classical multi-layer-perceptron (MLP) neural network model which takes the observations and coordinated action as input and produces an estimated score. The estimated score is then compared to the actual score reported from the environment.

In our experiments, we altered the training algorithm to suit the incremental expansion of the policy neural networks. The proposed training was carried out in two stages, action policy training and policy aggregation. We started by training two policy networks for 10000 steps (roughly 100 episodes). Both policies were initialised with different seeds in order to obtain diversity in the outcome. During the training the neural network model with the best score was stored. In the second stage, both action policies were combined and a new coordination policy was constructed. The coordination policy works as a switch to choose the weight of the actuation signals of different muscles and produce the final action. During this stage, the aggregated policies and the coordination policy were trained with a fresh experience replay buffer and a critic neural network. The rationale behind this is that the environment has changed from the policies' point of view and thus new experiences must be gathered.

When expanding the policy pool with a new policy network in the second stage, the trained weights of the previous policy network is copied to the new untrained policy network instead of the standard random initialisation. This allows the newly added policy to have a training head start with the current training state and protects the previously trained policies from being drastically altered while adapting to the newly added policy.

During the coordination refinement stage of the training, we prevent all policy networks from training and use the experience to fine tune the coordination network. This allows the coordination network to adapt to the distinctive postural stability strategies adopted by different policies. Locking the policies is also essential to preserve their trained strategies without leaking information from other policies and the coordinator network.

3. Results

During the training of the AI agent, the final goal was clearly defined by the scoring function implemented in the environment. However, in order to achieve this goal, the AI agent had to learn to discover and solve two more intermediate tasks,

1. identify the importance of centre of gravity (COG); and
2. identify and exploit the dominant leg concept.

The training took place on two stages with incrementally increasing number of policies from one policy towards a total of three policies.

As illustrated in Fig. 2-top, in the first few training episodes the AI agent explored the extremes of the de-innervating (10 episodes) and randomly innervating (100 episodes) the muscle-set controlling the body before it discovered the concept of the centre of gravity (COG). This allowed the AI agent attempt maintaining the COG and Zero-Moment-Point (ZMP) in front of the body. This, in return, allowed it to fall bottom-first instead of head-first. This resembles the behaviour toddlers exhibit when falling from a standing posture. It is worth noting that neither the COG nor the ZMP were provided as an input to the AI agent. In contrast to toddlers, the newly discovered concept (from the AI agent's perspective) was derived from the need to balance. Toddlers, on the other hand, already grasp this concept during the early stages of locomotion, which are sitting without assistance, crawling, standing with assistance and standing without assistance with an average of 1.43 ± 2.1 months between different milestones [51]. This is achieved via inputs from the vestibular, visual, and somatosensory sensory systems [52]. During the second stage of the training (two policies), the AI agent learnt to exploit the

concept of using a dominant leg [53]. This allowed it to shift the load of the entire body on one leg while using the other leg for fine tuning the balancing action.
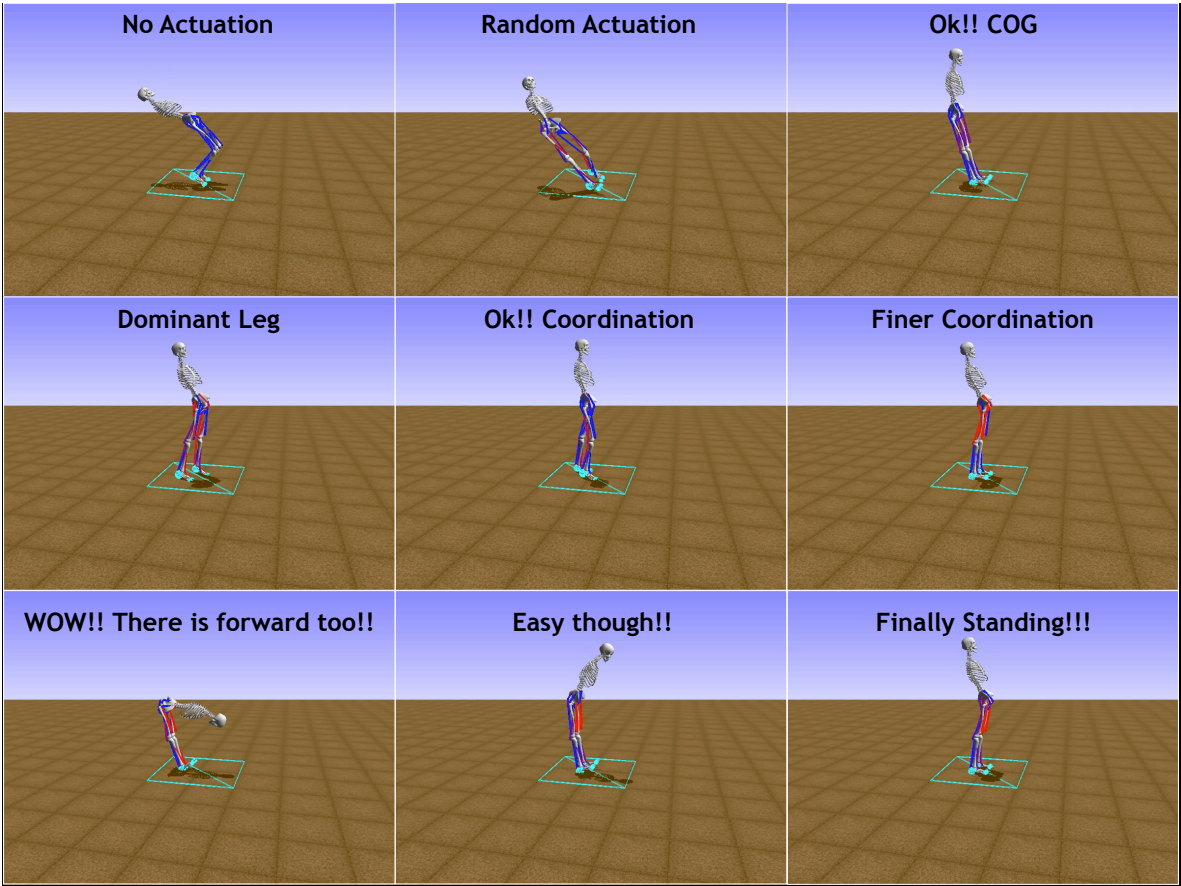
Figure 2. The learning of an AI agent towards maintaining a standing posture. Rows: Individually trained policies. Columns: Warm up episodes, better coordination, refined coordination (left to right). Muscles are coloured based on innervation level (Blue=0 to Red=1).
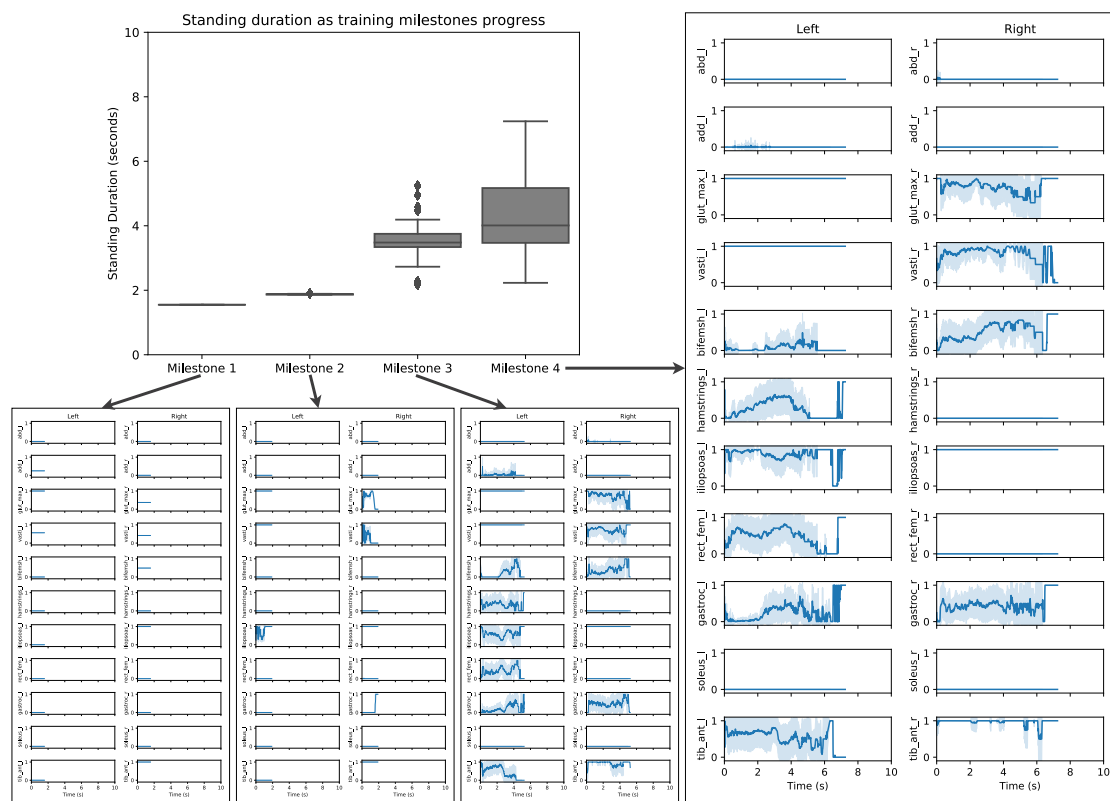
Figure 3. Four milestones of training (one trained model each). The box plot shows the duration before falling `epsdlen` for different milestones. The line plots show the AI-generated actuation signals for different left and right leg muscles. The AI agent adopted a knee locking strategy while thrusting the pelvis forward to control the centre of gravity. Each milestone was trained for 500000 simulation steps. The final agent was able to maintain balance for 4 seconds on average. The maximum training episode length was 500 simulation steps or 5 seconds (1 step=0.01 seconds). Solid blue line is the average of 35 test trials and the light blue envelop is the estimated standard error at $p < 0.05$ using random bootstrapping [54].

The first policy was trying to prevent the backward falls by exploiting the dominant leg and thrusting the pelvis forward. The second policy, however, adopted to perform a leaning forward action by pivoting on the heels and adjusting the ZMP via shifting the weight of the upper body anteriorly as shown in Fig. 2-bottom. While it abused the newly discovered capability, the fine tuning of the coordination neural network allowed it to maintain the balance between the new action and the previously learnt actions. Finally, during the fine tuning and coordination between the two trained policies, the AI agent explored the possibility of expanding the leg base and finally managed to stand.

As shown in Fig. 3, the muscle actuation pattern has changed in four milestones. Each milestone was trained for 500000 simulation steps. The maximum training episode length was 500 simulation steps or 5 seconds (1 step=0.01 seconds). Each milestone was evaluated via 35 test trials.[1] First, the AI agent adopted a strategy with three actuation levels (no, medium and full actuation). While this strategy does not actually maintain balance, it did serve as a foundation for subsequent milestones. In the next milestone, the AI agent adopted a left dominant leg strategy by maximising the actuation of the left gluteus maximus (`glut_max_l`) muscle to thrust the pelvis forward while locking the left knee with full extension by fully actuating the vastus intermedius (`vasti_l`) muscle. Accordingly, the right

---

[1]   Actuation tables are available in the supplementary materials.

gluteus maximus and vastus intermedius muscles were actuated to achieve dexterous traction with the ground.

This strategy was further refined in the third milestone which allowed the agent to prolong the balancing action by engaging the hamstrings (`hamstrings`) and the iliopsoas (`iliopsoas`) muscles for finer hip and knee control while exploiting alternating foot tapping as shown in the actuation graphs of the gastrocnemius (`gastroc`) and tibialis anterior (`tib_ant`) muscles. In the fourth and final milestone, the AI agent further improved the actuation strategy to maintain longer standing duration. Because the AI agent adopted a locked knee strategy, it did not attempt to actuate the soleus muscle (`soleus`). This can be considered a local minimum caused by the negligible weight of the effort penalty term in the score function (Eq. 3).

4. Discussion

Because each policy is self-contained, the proposed approach is expected to work with other off-policy reinforcement learning algorithms such as Soft Actor Critic (SAC) [55] and Distributed Distributional DDPG (D4PG) [56]. However, while early tests using SAC did show similar behaviour, we anticipate technical challenges with D4PG because of the asynchronous update to the experience replay buffer. In this section we discuss an interesting behaviour of the AI agent, limitations of the study, and future directions.

4.1. An Interesting Behaviour

An interesting AI behaviour emerged during an early training stage of the coordination neural network. Because the ultimate goal remains not to fall, the AI agent explored the option of taking a protective step to maintain a better balance. In doing so, the AI agent learnt to take few coordinated steps as shown in Fig. 4. This behaviour took place when we injected the AI agent with noise to increase exploration. However, because the score function was designed for standing, the learnt behaviour did not constitute a proper gait cycle. Also, the limited capacity of the AI model may have limited the dexterity of the learnt gait cycle. That been said, the ambition of the AI agent to engage in locomotion as a way to prevent falling remains an interesting behaviour. Considering Novacheck's conclusion that walking is controlled falling [4], the AI behaviour here cannot be considered walking because it is not self induced. This behaviour known as the *value misalignment problem* in computer science literature. This occurs when the score function is not specific enough to excite the AI agent to learn the desired task, but instead it causes the AI agent to engage in an obsessive behaviour of maximising the score by any means necessary. This problem poses a paradox because having a very specific score function may lead the neural networks to overfit on the observed training scenarios and fail to generalise to other variations in the environment. It is worth noting that this locomotion attempt (3 steps) could not be reproduced with the same coordination demonstrated in the video attached in the supplementary material.

4.2. Limitations

It is worth noting that, theoretically, the AI agent should be able to achieve the same result using only two policies (ideally a much deeper single policy). This limitation is usually addressed via model pruning which discards the redundant parts of the neural network. However, when applying model pruning, the performance (measured by standing duration) dropped by 50% and the model could not sustain a standing posture for more than two seconds. This suggests that the two policies do indeed contribute to posture stabilisation. We also noticed that after allowing training on a third policy, the AI agent discovered a new sub-task of slowly spreading the feet laterally to achieve a wider base. This behaviour opens the door for conducting further research into rearranging the policy neural networks into a chain or a pipeline. In this case, there will be no need for the coordination network. That being said, distributing the load over multiple neural networks does provide an explanation advantage of the behaviour of the AI model which is an important step towards explainable AI (XAI).
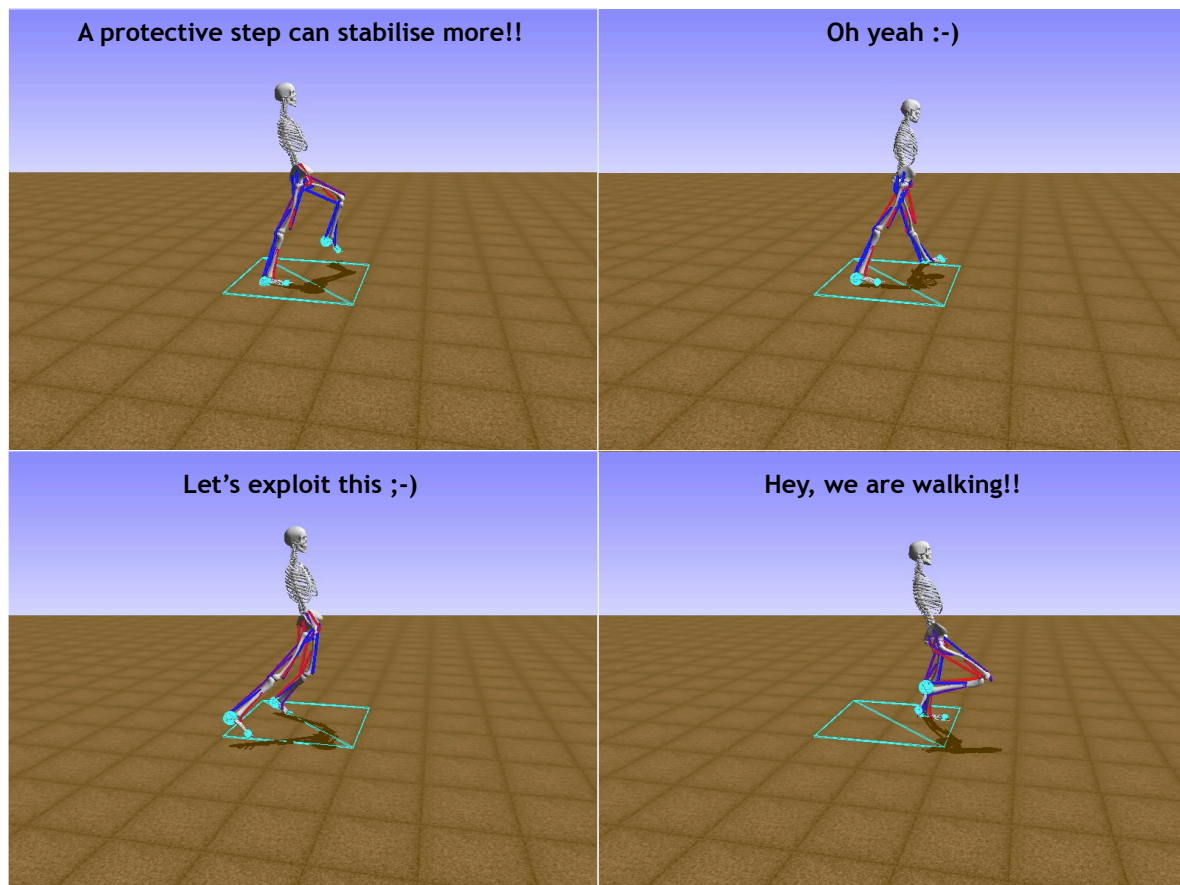
Figure 4. The AI agent explored the option of taking a protective step (top-left). When the associated score was positive (top-right), the AI agent started exploiting this with both legs (bottom-left). One thousand episodes later, the AI agent was able to provide coordinated actuation signals to perform a short locomotion (3 steps). Muscles are coloured based on innervation level (Blue=0 to Red=1). Locomotion video is available in the supplementary material.

The main challenge with training a single neural network on such a complex task is the lack of control over the flow of gradient update. Not only does it update the entire policy at once which alters the policy and the first few observation layers interacting with the environment, but also deprives the agent from perfecting any of the sub-tasks required to solve the problem. This was observed as an oscillation between two policies favouring leaning forward and backward in the 2D planar setup, i.e. no lateral movement. In the 3D setup, we used in this work, the lateral movement became a problem not only because of the added dimension but also because the hip adduction and hip abduction muscles are now engaged. These muscles' maximum isometric force is approximately 10 times the maximum isometric force of muscles like the biceps femoris which flexes the knee joint. This highlights the discrepancy in the characteristics of different actuators and is now being investigated using learnable parameterised activation functions in [57].

5. Conclusions

In this paper, we followed the learning journey of an AI agent attempting to assume a stable standing posture. We used the Opensim biomechanics simulation environment [30]. We adopted the DDPG reinforcement learning technique to derive coordinated continuous muscle actuation signals in order to stabilise a standing posture [34,49,50]. The AI agent learnt to maintain a standing posture for 4 seconds by learning two sub-tasks of leaning backward, forward and the coordination between the two actions. While considered a short duration for maintaining a standing posture, it is worth noting

that maintaining a standing posture for prolonged periods requires recurrent backtracking through different standing states. Such a recursive behaviour would require utilising Long-Short-Term-Memory modules (LSTM). Nevertheless, it was very interesting to witness the evolution of learnt sub-tasks as we allowed the AI agent to train on new policies.

The behaviour witnessed in this study highlights two more research points to be investigated. The first research point is related to training AI models using synthetic data. The main motivation deriving this area forward is the expanding gap in available data for training AI models. This issue becomes more significant when considering sensitive applications where collecting realistic data is difficult or may raise safety and ethical concerns. Fall detection and prevention is a growing concerns among public health where there is a shortage in datasets of realistic fall posture sequences. These datasets are usually recorded by stunt actors who can fall safely or generated by 3D artists [58]. However, both solutions do provide data that is not a real representation of fall occurrences. Fortunately, the presented work does derive the coordinated actuation of muscles that do cause a realistic fall. The AI agent's failed attempts to maintain a standing posture can provide a comprehensive dataset of falling posture sequences that can advance the fall detection and prevention research endeavours.

The second research point to be investigated is the discovered AI ambition to explore and exploit locomotion as a mean to prolong not falling. This problem is known in AI research as value misalignment problem and it has sparked a huge debate among computer scientists and philosophers. The reason for this is that relying on maximising the score solely can excite the AI agent to achieve this via exploiting the environment. While the argument of designing tighter score functions is sound, it is a very fine line that we have to walk between resorting to classical rule-based AI and the modern aspirations towards Artificial General Intelligence (AGI). This, in return, may introduce much feared scenarios of having intelligent machines running critical aspects of human lives [59]. While these scenarios are exaggerated in the media and the dystopian literature, it is unlikely to actually occur in the near future due to the limitation of compute power. However, not only does this debate raise good questions regarding AI safety, ethics and even rights, it also raises questions about our societal rights and duties.

Abbreviations

The following abbreviations are used in this manuscript:

AI	Artificial Intelligence
AGI	Artificial General Intelligence
ANN	Artificial Neural Network
COG/COM	Centre of Gravity/Mass
D4PG	Distributed Distributional Deep Deterministic Policy Gradient
DDPG	Deep Deterministic Policy Gradient
DoF	Degree of Freedom
DRL	Deep Reinforcement Learning
LSTM	Long-Short-Term-Memory
MLP	Multi-layer Perceptron
NEAT	Neuro-Evolution of Augmenting Topology
RL	Reinforcement Learning
XAI	Explainable Artificial Intelligence
ZMP	Zero Moment Point

References

1. Pua, Y.H.; Ong, P.H.; Clark, R.A.; Matcher, D.B.; Lim, E.C.W. Falls efficacy, postural balance, and risk for falls in older adults with falls-related emergency department visits: prospective cohort study. *BMC geriatrics* **2017**, *17*, 291.

2. Alonso, A.C.; Brech, G.C.; Bourquin, A.M.; Greve, J.M.D. The influence of lower-limb dominance on postural balance. *Sao Paulo Medical Journal* **2011**, *129*, 410–413.

3. Riemann, B.L.; Guskiewicz, K.M. *Contribution of the peripheral somatosensory system to balance and postural equilibrium*; 2000; pp. 37–51.

4. Novacheck, T.F. The biomechanics of running. *Gait & posture* **1998**, *7*, 77–95.

5. Winter, D.A.; Quanbury, A.O.; Reimer, G.D. Analysis of instantaneous energy of normal gait. *Journal of biomechanics* **1976**, *9*, 253–257.

6. Miller, C.A.; Verstraete, M.C. A mechanical energy analysis of gait initiation. *Gait & posture* **1999**, *9*, 158–166.

7. Inman, V.T.; Ralston, H.J.; Todd, F. *Human walking*; Williams & Wilkins, 1981.

8. Devine, J. The versatility of human locomotion. *American Anthropologist* **1985**, *87*, 550–570.

9. Winter, D.A. *Biomechanics and motor control of human movement*; John Wiley & Sons, 2009.

10. Bohannon, R.W. Comfortable and maximum walking speed of adults aged 20—79 years: reference values and determinants. *Age and ageing* **1997**, *26*, 15–19.

11. Serra-AÑó, P.; López-Bueno, L.; García-Massó, X.; Pellicer-Chenoll, M.T.; González, L.M. Postural control mechanisms in healthy adults in sitting and standing positions. *Perceptual and motor skills* **2015**, *121*, 119–134.

12. Park, E.; Reimann, H.; Schöner, G. Coordination of muscle torques stabilizes upright standing posture: an UCM analysis. *Experimental brain research* **2016**, *234*, 1757–1767.

13. Barroso, F.O.; Torricelli, D.; Molina-Rueda, F.; Alguacil-Diego, I.M.; Cano-de-la Cuerda, R.; Santos, C.; Moreno, J.C.; Miangolarra-Page, J.C.; Pons, J.L. Combining muscle synergies and biomechanical analysis to assess gait in stroke patients. *Journal of Biomechanics* **2017**, *63*, 98–103.

14. Seth, A.; Hicks, J.L.; Uchida, T.K.; Habib, A.; Dembia, C.L.; Dunne, J.J.; Ong, C.F.; DeMers, M.S.; Rajagopal, A.; Millard, M.; others. OpenSim: Simulating musculoskeletal dynamics and neuromuscular control to study human and animal movement. *PLoS computational biology* **2018**, *14*, e1006223.

15. Raabe, M.E.; Chaudhari, A.M. An investigation of jogging biomechanics using the full-body lumbar spine model: Model development and validation. *Journal of biomechanics* **2016**, *49*, 1238–1243.

16. Delp, S.L.; Anderson, F.C.; Arnold, A.S.; Loan, P.; Habib, A.; John, C.T.; Guendelman, E.; Thelen, D.G. OpenSim: open-source software to create and analyze dynamic simulations of movement. *IEEE transactions on biomedical engineering* **2007**, *54*, 1940–1950.

17. Reinbolt, J.A.; Seth, A.; Delp, S.L. Simulation of human movement: applications using OpenSim. *Procedia IUTAM* **2011**, *2*, 186–198.

18. Iskander, J.; Hossny, M.; Nahavandi, S. Using biomechanics to investigate the effect of VR on eye vergence system. *Applied Ergonomics* **2019**, *81*, 102883. doi:10.1016/j.apergo.2019.102883.

19. Iskander, J.; Hossny, M.; Nahavandi, S.; Del Porto, L. An Ocular Biomechanic Model for Dynamic Simulation of Different Eye Movements. *Journal of biomechanics* **2018**, *71*, 208–216.

20. Nahavandi, D.; Iskander, J.; Hossny, M.; Haydari, V.; Harding, S. Ergonomic effects of using Lift Augmentation Devices in mining activities. Systems, Man, and Cybernetics (SMC), 2016 IEEE International Conference on. IEEE, 2016, pp. 002012–002019.

21. Abobakr, A.; Nahavandi, D.; Hossny, M.; Iskander, J.; Attia, M.; Nahavandi, S.; Smets, M. RGB-D ergonomic assessment system of adopted working postures. *Applied Ergonomics* **2019**, *80*, 75–88.

22. Hossny, M.; Nahavandi, D.; Nahavandi, S.; Haydari, V.; Harding, S. Musculoskeletal analysis of mining activities. Systems Engineering (ISSE), 2015 IEEE International Symposium on. IEEE, 2015, pp. 184–189.

23. Dorn, T.W.; Wang, J.M.; Hicks, J.L.; Delp, S.L. Predictive simulation generates human adaptations during loaded and inclined walking. *PloS one* **2015**, *10*, e0121407.

24. DeMers, M.S.; Hicks, J.L.; Delp, S.L. Preparatory co-activation of the ankle muscles may prevent ankle inversion injuries. *Journal of biomechanics* **2017**, *52*, 17–23.

25. Halilaj, E.; Rajagopal, A.; Fiterau, M.; Hicks, J.L.; Hastie, T.J.; Delp, S.L. Machine learning in human movement biomechanics: best practices, common pitfalls, and new opportunities. *Journal of biomechanics* **2018**, *81*, 1–11.

26. Haarnoja, T.; Ha, S.; Zhou, A.; Tan, J.; Tucker, G.; Levine, S. Learning to walk via deep reinforcement learning. *arXiv preprint arXiv:1812.11103* **2018**.

27. Hebbel, M.; Kosse, R.; Nistico, W. Modeling and learning walking gaits of biped robots. Proceedings of the Workshop on Humanoid Soccer Robots of the IEEE-RAS International Conference on Humanoid Robots, 2006, pp. 40–48.

28. Kober, J.; Bagnell, J.A.; Peters, J. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research* **2013**, *32*, 1238–1274.

29. Sutton, R.S.; Barto, A.G.; others. *Introduction to reinforcement learning*; Vol. 2, MIT press Cambridge, 1998.

30. Kidziński, Ł.; Ong, C.; Mohanty, S.P.; Hicks, J.; Carroll, S.; Zhou, B.; Zeng, H.; Wang, F.; Lian, R.; Tian, H.; others. Artificial Intelligence for Prosthetics: Challenge Solutions. In *The NeurIPS'18 Competition*; Springer, 2020; pp. 69–128.

31. Kidziński, Ł.; Mohanty, S.P.; Ong, C.F.; Hicks, J.L.; Carroll, S.F.; Levine, S.; Salathé, M.; Delp, S.L. Learning to run challenge: Synthesizing physiologically accurate motion using deep reinforcement learning. In *The NIPS'17 Competition: Building Intelligent Systems*; Springer, 2018; pp. 101–120.

32. Kidziński, Ł.; Mohanty, S.P.; Ong, C.F.; Huang, Z.; Zhou, S.; Pechenko, A.; Stelmaszczyk, A.; Jarosik, P.; Pavlov, M.; Kolesnikov, S.; others. Learning to Run challenge solutions: Adapting reinforcement learning methods for neuromusculoskeletal environments. In *The NIPS'17 Competition: Building Intelligent Systems*; Springer, 2018; pp. 121–153.

33. Jaśkowski, W.; Lykkebø, O.R.; Toklu, N.E.; Trifterer, F.; Buk, Z.; Koutník, J.; Gomez, F. Reinforcement Learning to Run... Fast. In *The NIPS'17 Competition: Building Intelligent Systems*; Springer, 2018; pp. 155–167.

34. Lillicrap, T.P.; Hunt, J.J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; Wierstra, D. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971* **2015**.

35. Schulman, J.; Moritz, P.; Levine, S.; Jordan, M.; Abbeel, P. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438* **2015**.

36. Levine, S.; Finn, C.; Darrell, T.; Abbeel, P. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research* **2016**, *17*, 1334–1373.

37. Duan, Y.; Chen, X.; Houthooft, R.; Schulman, J.; Abbeel, P. Benchmarking deep reinforcement learning for continuous control. International Conference on Machine Learning, 2016, pp. 1329–1338.

38. Pilotto, A.; Boi, R.; Petermans, J. Technology in geriatrics. *Age and ageing* **2018**, *47*, 771–774.

39. Lee, S.; Lee, K.; Park, M.; Lee, J. Scalable Muscle-actuated Human Simulation and Control. *ACM Transactions on Graphics* **2019**, *37*.

40. Brockman, G.; Cheung, V.; Pettersson, L.; Schneider, J.; Schulman, J.; Tang, J.; Zaremba, W. Openai gym. *arXiv preprint arXiv:1606.01540* **2016**.

41. Delp, S.L.; Loan, J.P.; Hoy, M.G.; Zajac, F.E.; Topp, E.L.; Rosen, J.M. An interactive graphics-based model of the lower extremity to study orthopaedic surgical procedures. *IEEE Transactions on Biomedical engineering* **1990**, *37*, 757–767.

42. Ong, C.F.; Geijtenbeek, T.; Hicks, J.L.; Delp, S.L. Predictive simulations of human walking produce realistic cost of transport at a range of speeds. Proceedings of the 16th International Symposium on Computer Simulation in Biomechanics, 2017, pp. 19–20.

43. Arnold, E.M.; Ward, S.R.; Lieber, R.L.; Delp, S.L. A model of the lower limb for analysis of human movement. *Annals of biomedical engineering* **2010**, *38*, 269–279.

44. Thelen, D.G. Adjustment of muscle mechanics model parameters to simulate dynamic contractions in older adults. *Journal of biomechanical engineering* **2003**, *125*, 70–77.

45. Hunt, K.; Crossley, F. Coefficient of Restitution Interpreted as Damping in Vibroimpact. *Journal of Applied Mechanics* **1975**, *42*, 440–445.

46. Stanley, K.O.; Miikkulainen, R. Evolving Neural Networks through Augmenting Topologies. *Evolutionary Computation* **2002**, *10*, 99–127.

47. Stanley, K.O. Neuroevolution: A different kind of deep learning. *Obtenido de https://www. oreilly. com/ideas/neuroevolution-a-different-kindof-deep-learning el* **2017**, *27*, 2019.

48. Hospedales, T.; Antoniou, A.; Micaelli, P.; Storkey, A. Meta-Learning in Neural Networks: A Survey, 2020, [arXiv:cs.LG/2004.05439].

49. Grondman, I.; Busoniu, L.; Lopes, G.A.; Babuska, R. A survey of actor-critic reinforcement learning: Standard and natural policy gradients. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* **2012**, *42*, 1291–1307.

50. Konda, V.R.; Tsitsiklis, J.N. Actor-critic algorithms. Advances in neural information processing systems, 2000, pp. 1008–1014.

51. Ghassabian, A.; Sundaram, R.; Bell, E.; Bello, S.C.; Kus, C.; Yeung, E. Gross Motor Milestones and Subsequent Development **2016**. *138*. doi:10.1542/peds.2015-4372.

52. Forbes, P.A.; Chen, A.; Blouin, J.S. Sensorimotor control of standing balance. In *Handbook of clinical neurology*; Elsevier, 2018; Vol. 159, pp. 61–83.

53. Spry, S.; Zebas, C.; Visser, M. What is leg dominance? ISBS-Conference Proceedings Archive, 1993.

54. Dragicevic, P. Fair Statistical Communication in HCI. In *Modern Statistical Methods for HCI*; Springer, 2016; pp. 291 – 330. doi:10.1007/978-3-319-26633-6_13.

55. Haarnoja, T.; Zhou, A.; Abbeel, P.; Levine, S. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor, 2018, [arXiv:cs.LG/1801.01290].

56. Barth-Maron, G.; Hoffman, M.W.; Budden, D.; Dabney, W.; Horgan, D.; TB, D.; Muldal, A.; Heess, N.; Lillicrap, T. Distributed Distributional Deterministic Policy Gradients, 2018, [arXiv:cs.LG/1804.08617].

57. Hossny, M.; Iskander, J.; Attia, M.; Saleh, K. Refined Continuous Control of DDPG Actors via Parametrised Activation, 2020, [arXiv:cs.LG/2006.02818].

58. Abobakr, A.; Hossny, M.; Nahavandi, S. A Skeleton-Free Fall Detection System From Depth Images Using Random Decision Forest. *IEEE Systems Journal* **2018**, *PP*, 1–12. doi:10.1109/JSYST.2017.2780260.

59. Rahwan, I.; Cebrian, M.; Obradovich, N.; Bongard, J.; Bonnefon, J.F.; Breazeal, C.; Crandall, J.W.; Christakis, N.A.; Couzin, I.D.; Jackson, M.O.; Jennings, N.R.; Kamar, E.; Kloumann, I.M.; Larochelle, H.; Lazer, D.; McElreath, R.; Mislove, A.; Parkes, D.C.; Pentland, A.S.; Roberts, M.E.; Shariff, A.; Tenenbaum, J.B.; Wellman, M. Machine behaviour. *Nature* **2019**, *568*, 477–486. doi:10.1038/s41586-019-1138-y.