Neural Networks 23 (2010) 587–591

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Neural networks letter

Adaptive critics for dynamic optimization

Raghavendra V. Kulkarni *, Ganesh Kumar Venayagamoorthy

Real-Time Power and Intelligent Systems Laboratory, Missouri University of Science and Technology, Rolla, MO, USA

ARTICLE INFO

Article history:
Received 28 December 2009
Accepted 3 February 2010

Keywords:
Adaptive critic design
Energy efficiency
Sleep scheduling
Wildlife monitoring
Wireless sensor networks

ABSTRACT

A novel action-dependent adaptive critic design (ACD) is developed for dynamic optimization. The proposed combination of a particle swarm optimization-based actor and a neural network critic is demonstrated through dynamic sleep scheduling of wireless sensor motes for wildlife monitoring. The objective of the sleep scheduler is to dynamically adapt the sleep duration to node's battery capacity and movement pattern of animals in its environment in order to obtain snapshots of the animal on its trajectory uniformly. Simulation results show that the sleep time of the node determined by the actor critic yields superior quality of sensory data acquisition and enhanced node longevity.

1. Introduction

Adaptive critic designs (ACDs) are popular in neurocontrol applications due to their ability to learn in noisy, nonlinear, and non-stationary environments. The adaptive critic method determines optimal control laws for a system by adapting two subsystems, an actor (which dispenses control signals) and a critic (which learns the desired performance index for some function associated with the performance index) (Prokhorov & Wunsch, 1997). These subsystems together approximate the Hamilton–Jacobi–Bellman equation associated with optimal control theory (Si, Barto, Powell, & Wunsch, 2004). A novel action-dependent heuristic dynamic programming (ADHDP) type ACD suitable for dynamic optimization is introduced in this paper. This ACD uses a particle swarm optimization (PSO) based actor (del Valle, Venayagamoorthy, Mohagheghi, Hernandez, & Harley, 2008), whose performance is evaluated by a critic neural network. The effectiveness of the proposed ACD is demonstrated by means of dynamic optimization of sleep duration in a sensor node deployed for wildlife monitoring. Besides PSO- and neural-networks-based actor networks, dynamic logic can be investigated for dynamic optimization (Perlovsky, 2009a, 2009b; Perlovsky & Deming, 2007).

Field biologists are studying wildlife habitat, migration and foraging patterns using wireless sensor networks (WSNs) (Trifa, Girod, Collier, Blumstein, & Taylor, 2007). WSNs deployed for wildlife monitoring are networks of battery-powered sensor motes. To conserve energy, motes remain in sleep mode most of the time, and wake up periodically to acquire sensory data. Sleep cycles of a constant duration are unsuitable to wildlife monitoring due to unpredictable animal behavior. Short sleep cycles result in fast battery depletion, and long sleep cycles result in motes that miss animal activities, resulting in low quality of acquired sensory data. This necessitates dynamic sleep scheduling. The PSO-based ADHDP is used in this study for dynamic sleep scheduling.

The mission field of the network of sensor motes having the proposed ACD-based dynamic sleep schedulers is shown in Fig. 1. The task of the WSN is to record trajectories of wild animals by capturing its snapshots at uniform distances. It is assumed that WSN coverage and connectivity requirements are met (Bai, Kumar, Xuan, Yun, & Lai, 2006). Each node is assumed to have a sensor, through which it accurately estimates the position of an animal within its sensing range. In a sleep cycle k, each node wakes up and remains active for a period of T_{on} seconds. While in active state, if an animal has moved into the node's sensing radius R_s meters, the node uses the ACD-based sleep scheduler to determine its sleep time $T(k)$. If not, it goes to sleep for a quiescent duration of T_Q seconds as shown in Fig. 2(a). It is desired that the sleep time should adapt to battery capacity $E(k)$ of the node, and the speed $S(k)$ of the animal. For good quality of data acquisition, it is necessary that the new position of the animal is recorded before the animal moves not greater than the resolution distance D_r meters. If the animal moves at instantaneous speed of S_Q km/h or slower, the node can record positions of the animal at every D_r meters and still sleep for the quiescent duration of T_Q seconds. If the animal moves faster, in order to acquire good sensory data, its instantaneous sleep duration $T(k)$ has to decrease to an optimal value $T^*(k)$ as expressed in (1) (see Fig. 2(b)).

This is a problem with two objectives, which are weighted and combined into one. Good acquisition quality and resilience to battery capacity are given weights of α and β respectively, with

* Corresponding author.
E-mail addresses: arvie@ieee.org (R.V. Kulkarni), gkumar@ieee.org (G.K. Venayagamoorthy).

0893-6080/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.neunet.2010.02.002

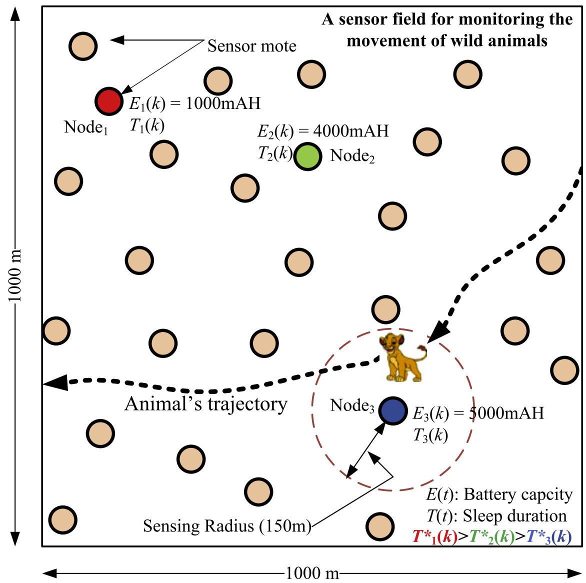


Fig. 1. A WSN for monitoring wild animals.

$\alpha + \beta = 1.$

$$T(k) = \begin{cases} T_Q & \text{if } S(k) \le S_Q \\ T^*(k) & \text{Otherwise.} \end{cases} \tag{1}$$

2. Dynamic sleep scheduler

The mutually conflicting requirements of long network life and good sensing quality are addressed by the PSO-actor-based ADHDP type ACD. Interconnections between the actor and the critic in the proposed design are shown in Fig. 3. The actor uses PSO to determine sleep duration. The critic approximates the Hamilton–Jacobi–Bellman equation, and provides scalar value as a performance feedback. This helps the actor to modify its output in order to achieve the set goal.

PSO is an iterative algorithm that models communal behavior of a biological species (del Valle et al., 2008). It uses primitive mathematical operators, and is inexpensive in storage and computational requirements, thus suitable to be implemented on resource constrained computing platforms such as WSN motes.

PSO consists of a population of s candidate solutions called particles that move in a d-dimensional solution hyperspace in search of the global solution. In the current problem, the number of parameters to be determined is $d = 1$. Each particle j, $1 \le j \le s$, represents a $T(k)_j$, the sleep duration in the sleep cycle k. Each particle j occupies position $T(k)_j$ and moves with velocity v_j. Particles are initially assigned random positions and velocities within fixed boundaries. In each iteration i, fitness of each particle is evaluated by an objective function $f(T(k), i)$. In the global-best version of PSO, each particle j stores $T_p(k)_j$, the position where it achieved its best fitness, and $T_g(k)$, the position of the current fittest particle.

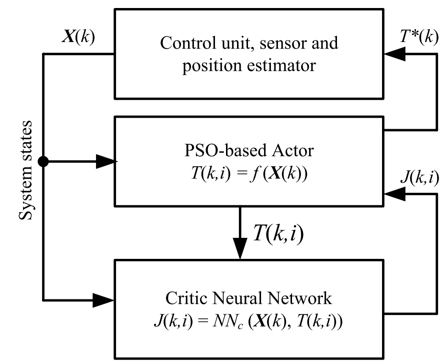


Fig. 3. General block diagram of the ADHDP dynamic sleep scheduler (sleep cycle k, PSO iteration i).

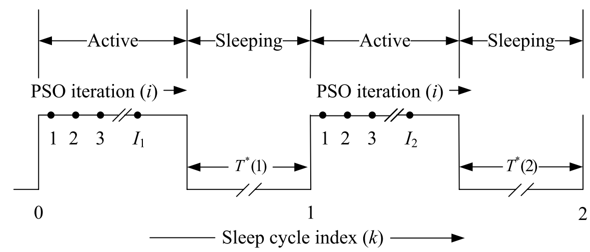


Fig. 4. The relationship between sleep cycle index k and PSO iteration i.

In each iteration i, velocity v_j and position $T(k)_j$ of jth particle are updated using (2) and (3).

$$v_j^{i+1} = wv_j^i + \varphi_1 r_1 (T_p(k)_j - T(k)_j) + \varphi_2 r_2 (T_g(k) - T(k)_j) \tag{2}$$

$$T(k)_j^{i+1} = T(k)_j^i + v_j^{i+1}. \tag{3}$$

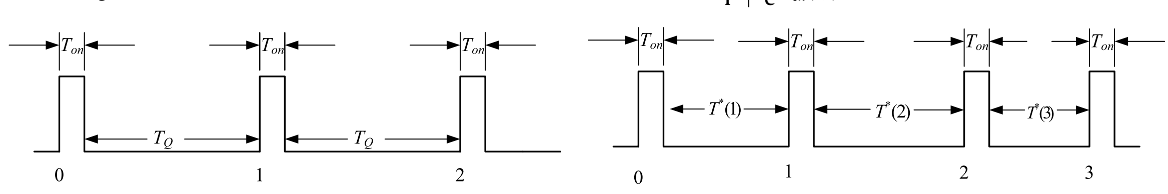
Here w, φ_1 and φ_2 are constants, and r_1 and r_2 are random numbers distributed uniformly in [0,1]. $T_g(k)$ at the end of an iteration represents the best solution achieved so far. PSO is terminated either if the desired fitness is achieved or if the maximum number of allowable iterations $i_{\max}$ is reached. The final $T_g(k)$ is taken as the optimal sleep duration $T^*(k)$ (del Valle et al., 2008). The relation between sleep cycle index k and PSO iteration i is shown in Fig. 4.

The structure of the neural network used as the critic is shown in Fig. 5. The output of the critic $J(k, i)$, an approximation for weighted total future reward-to-go, in the ith iteration of PSO in kth sleep cycle is obtained as in (4).

$$J(k, i) = \sum_{n=1}^{N_H} w_n^{(o)} p_n(k, i). \tag{4}$$

Here $w_n^{(o)}$ is the weight associated with $p_n(k, i)$, the output of the nth neuron in the hidden later, and $N_H = 8$ is the number of hidden layer neurons. $p_n(k, i)$ is computed as in (5).

$$p_n(k, i) = \frac{1 - e^{-q_n(k,i)}}{1 + e^{-q_n(k,i)}}, \quad n = 1, 2, \ldots, N_H \tag{5}$$



(a) Animal's speed $S(k) \le SQ$ and $E(k) = EF$.

(b) Animal's speed $S(k) > SQ$ or $E(k) \ne EF$.

Fig. 2. Adaptive sleep schedule.

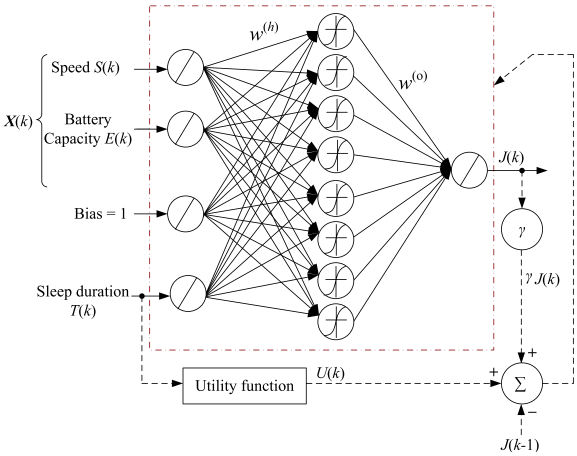


Fig. 5. The structure and training of the critic neural network.

$$q_n(k,i) = \sum_{j=1}^{N_I} w_{jn}^{(h)} x_j(k,i), \quad j = 1, 2, \ldots, N_I. \tag{6}$$

The input to a neuron n in the hidden layer at time instance k, $q_n(k)$, is computed as in (6), where $w_{jn}^{(h)}$ is the weight associated with the connection from jth neuron in input layer to nth neuron in hidden layer, and $N_I = 4$ is the number of input layer neurons. The critic receives four inputs, namely, $S(k)$, $V(k)$, $T(k, i)$ and a constant bias as shown in Fig. 5. The critic neural network is trained using backpropagation algorithm. The training error for the critic in a training iteration t is $E_c(t) = \frac{1}{2}e_c(t)$, where $e_c(t)$ is computed as in (7), which is shown in dotted lines in Fig. 5.

$$e_c(t) = [U(t) + \gamma J(t)] - J(t-1). \tag{7}$$

Here, $0 \le \gamma \le 1$ is the discount factor. Weights in output layer and hidden layer are updated as (8) and (9) respectively (Si et al., 2004).

$$w_n^{(o)}(t+1) = w_n^{(o)}(t) + \Delta w_n^{(o)} \tag{8}$$

$$w_{jn}^{(h)}(t+1) = w_{jn}^{(h)}(t) + \Delta w_{jn}^{(h)}. \tag{9}$$

The reward function or the local utility function $U(T(t))$ is defined as in (10).

$$U(t) = \alpha\,|1 - \Delta D(t)T(t)\theta| + \beta\,|1 - E(t)T(t)\phi| + 100(D(t) > R_s)(T(t) \ne T_Q). \tag{10}$$

Here, θ and ϕ are constants computed as (11) and (12) respectively, and $\Delta D(t)$ is the distance covered by the animal in the time cycle t.

$$\theta = \frac{1}{D_r \cdot (T_Q + T_{on})} \tag{11}$$

$$\phi = \frac{1}{E_{full} \cdot (T_Q + T_{on})}. \tag{12}$$

E_{full} is the capacity of the charged battery.

The data pertaining to instantaneous speed of the animal, the battery capacity, and the sleep time determined by the actor in 1000 iterations of PSO is collected, and used for off-line training the critic for 500 iterations of backpropagation. During the normal operation of the WSN, PSO thrives to minimize $J(k)$. In a sleep cycle k the actor outputs the sleep duration determined in the previous cycle $k-1$. PSO search is conducted until either the value of $J(k, i)$ is acceptably low, or a maximum number of iterations is reached, whichever occurs earlier. Therefore, it is not necessary that PSO should be run in each sleep cycle. Besides, the number of PSO iterations is not necessarily the same in all sleep cycles.

3. Results

The test node and animal movement are simulated in Matlab. The wake-up time in each cycle $T_{on} = 0.5$ s. The resolution distance $D_r = 10$ m. The simulated animal moves with a mean speed S_M across the sensing range of the node. In every sleep cycle k, it changes its direction with a probability 0.2, and changes its instantaneous speed $S(k)$ randomly, within the limits of $S_M \pm 0.1S_M$. It is assumed that $E_{full} = 5400$ mAH, the capacity of two fully charged AA sized batteries. The node draws currents of 8 mA and 8 μA in active and sleep states respectively. This energy model complies with the Crossbow IRIS mote. PSO constants w, φ_1 and φ_2 are set to 0.9, 2.0 and 2.0 respectively.

The dynamic sleep durations determined by ADHDP in a trial run are shown in Fig. 6. When the animal moves into the sensing range, and if its speed is equal or less than S_Q, its sleep duration is constant, $T_Q = 4.0$ s. But, if the instantaneous speed of the animal increases, the sleep duration adapts to the variation in speed and node's battery capacity. The mean D_μ and the standard deviation D_σ of the distances covered by the animal in a sleep cycle over all data samples on the animal's trace reflect the quality of acquisition. The adaptive sleep durations determined by the ADHDP-based scheduler, and the distance covered by the animal in each sleep cycle in the same trial run are shown in Fig. 7.

A summary of 30 trial runs for different animal speeds S_M and different settings of initial battery capacity E_{ini} and different combinations of α and β is given in Table 1. With constant sleep duration T_Q in absence of the ACD, the mean change in battery capacity ΔE_μ is lower than in the other two cases. However, the resulting data acquisition quality is poorer. With $\alpha = 1$ and $\beta = 0$, the sleep duration is not resilient the battery capacity. Therefore, ΔE_μ depends only on the animal's speed. With $\alpha = 0.9$ and $\beta = 0.1$, the sleep duration adapts to both the speed and battery capacity. It can be noted that ΔE_μ reduces with battery capacity at the cost of data acquisition quality. Lower value of ΔE_μ denotes lower energy expenditure, and thus, longer node life.

4. Conclusion

A novel action-dependent ACD is introduced for dynamic optimization and its effectiveness is demonstrated through

Table 1
Higher quality of data acquisition through ACD-based dynamic sleep scheduling.

S_M (km/h)	E_{ini} (mAH)	Without ACD ($T(k) = 3.1$ s $\forall k$)			With ACD ($\alpha = 1,\ \beta = 0$)			With ACD ($\alpha = 0.9,\ \beta = 0.1$)		
		D_μ (m)	D_σ	ΔE_μ (mAH)	D_μ (m)	D_σ	ΔE_μ (mAH)	D_μ (m)	D_σ	ΔE_μ (mAH)
10	5400	10.12	0.27	0.0335	10.22	0.79	0.0374	10.22	0.79	0.0374
	1000	10.12	0.27	0.0335	10.22	0.79	0.0374	10.98	0.62	0.0331
15	5400	15.07	0.14	0.0224	10.23	0.46	0.0482	10.23	0.46	0.0482
	1000	15.07	0.14	0.0224	10.23	0.46	0.0482	10.81	0.28	0.0422
20	5400	20.04	0.09	0.0168	10.01	0.19	0.0394	10.01	0.19	0.0394
	1000	20.04	0.09	0.0168	10.01	0.19	0.0394	10.53	0.27	0.0367
25	5400	25.17	0.07	0.0134	10.01	0.22	0.0330	10.01	0.22	0.0330
	1000	25.17	0.07	0.0134	10.01	0.22	0.0330	10.33	0.35	0.0318

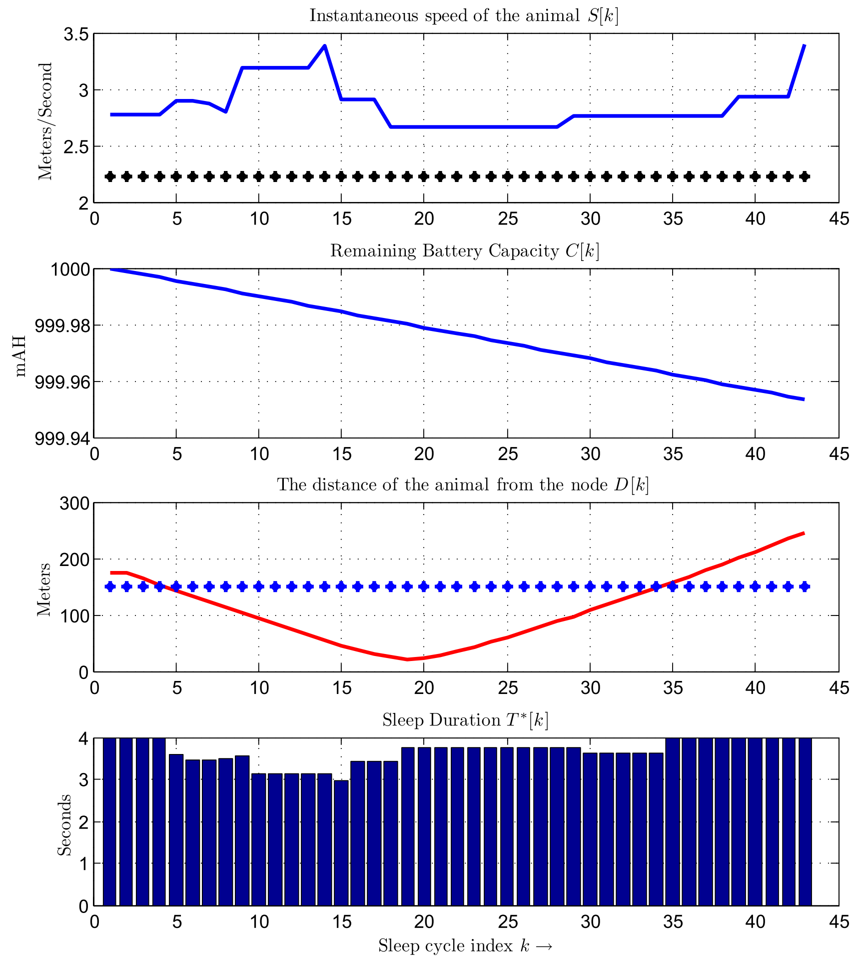


Fig. 6. Speed profile, sleep duration, battery capacity and the distance profile of the animal (Mean speed $S_M = 10$ km/h, $E_{ini} = 1000$ mAH).

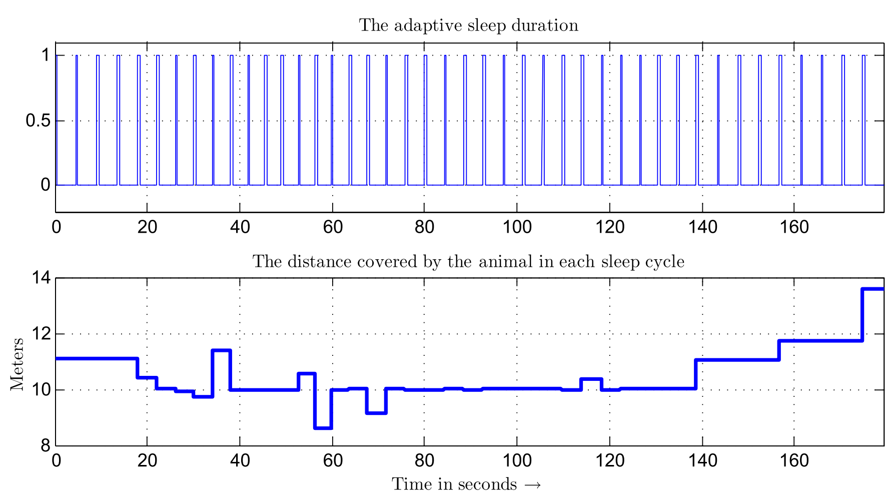


Fig. 7. The optimal sleep durations computed by the ACD scheduler and the distance covered by the animal in each time cycle.

dynamic sleep scheduling of a sensor mote for wildlife monitoring. The ACD has a PSO-based actor and a neural network critic. Dynamic logic in the ACD framework remains to be investigated. The dynamic sleep schedule determined by the ACD results in high quality of data acquisition and improved energy efficiency. A potential direction for extension of this study is to investigate lightweight forms of PSO and compact neuron structures. Besides, there exists a scope for investigating the application of the ACD in multi-node collaborative sleep scheduling.

Acknowledgements

The authors gratefully acknowledge the financial support for this work provided by the National Science Foundation, USA, under the grants *ECCS* #0529292, *ECCS* #0348221 *and EFRI* #0836017.

References

Bai, X., Kumar, S., Xuan, D., Yun, Z., & Lai, T. H. (2006). Deploying wireless sensors to achieve both coverage and connectivity. In *Proceedings of the 7th ACM international symposium on mobile ad hoc networking and computing* (pp. 131–142). New York, USA: ACM.

del Valle, Y., Venayagamoorthy, G. K., Mohagheghi, S., Hernandez, J. C., & Harley, R. G. (2008). Particle swarm optimization: Basic concepts, variants and applications in power systems. *IEEE Transactions on Evolutionary Computation*, *12*(2), 171–195.

Perlovsky, L. (2009a). Language and cognition. *Neural Networks*, *22*(3), 247–257. Goal-Directed Neural Systems.

Perlovsky, L. (2009b). Vague-to-crisp neural mechanism of perception. *IEEE Transactions on Neural Networks*, *20*(8), 1363–1367.

Perlovsky, L. I., & Deming, R. W. (2007). Neural networks for improved tracking. *IEEE Transactions on Neural Networks*, *18*(6), 1854–1857.

Prokhorov, D., & Wunsch, D. (1997). Adaptive critic designs. *IEEE Transactions on Neural Networks*, *8*(5), 997–1007.

Si, J., Barto, A., Powell, W., & Wunsch, D. (2004). *IEEE press series on computational intelligence*, *Handbook of learning and approximate dynamic programming*. Wiley–IEEE Press.

Trifa, V., Girod, L., Collier, T., Blumstein, D., & Taylor, C. (2007). Automated wildlife monitoring using self-configuring sensor networks deployed in natural habitats. In *International symposium on artificial life and robotics (AROB07)*. Beppu, Japan.