NEURO-DYNAMIC PROGRAMMING FOR THE EXPLORATION OF UNKNOWN GRAPHS

M. Baglietto G. Battistelli L. Scardovi R. Zoppoli

Department of Communications, Computer,
and System Sciences, DIST–University of Genoa,
Via Opera Pia 13, 16145 Genova, Italy.
E-mail: {mbaglietto,bats,lucas,rzop}@dist.unige.it

Abstract: In this paper, the problem of exploring stochastic graphs is addressed. The definition of the *entropy* related to the a-priori unknown parameters (the lengths of the a-priori unknown links) leads to the formulation of the problem as a stochastic optimal control one. The application of exact Dynamic Programming suffers the so-called curse of dimensionality. To overcome this drawback, an approximate technique is proposed making use of Neuro-Dynamic Programming. Exploiting the concept of *frontier*, any approximate solution of the problem is shown to generate a "proper" policy. *Copyright© 2005 IFAC*

Keywords: Neural networks, Dynamic programming, Graph, Control, Entropy.

1. INTRODUCTION

In this paper we consider the problem of exploring an a-priori partially unknown graph. This general problem can model a number of interesting decisional and control problems such as extra-planetary robotic exploration, see floor mapping, search for free channels in communication networks, etc. The graph exploration can be formulated as a problem in which a controller (or Decision Maker-DM) must acquire all the information about a finite set of unknown parameters. From this point of view, the problem considered in this paper is the discrete space version of previous works on the *active identification problem* (Baglietto *et al.*, 2003*b*; Baglietto *et al.*, 2004) in which the general problem is stated.

One of the first exploration problems formulated on graphs was the *on-line chinese postman problem* (Deng and Papadimitriou, 1999): an agent must traverse all edges of an unknown, directed and strongly connected graph and return to the starting vertex. For this problem the authors introduced some heuristic algorithms and gave upper and lower bounds for them. A similar model for the robotic exploration problem can be found in (Baglietto *et al.*, 2003*b*).

We address the problem in a very general setting, where the DM sensors are affected by noise. In this case the problem can be formulated as a stochastic optimal control problem on a continuous but finite dimensional state space. Moreover, under a particular assumption, it is possible to consider a discrete model in time and space and, by using the concept of entropy, to reformulate the problem as a stochastic shortest path problem (see (Bertsekas and Tsitsiklis, 1996)).

By exploiting the concept of "frontier nodes" an equivalent formulation to the original problem is given, for which any policy is a "proper" one and the Dynamic Programming (DP) value iteration algorithm converges in a finite number of steps. The complexity of the problem dealt with leads us to consider techniques similar to Neuro-Dynamic Programming (NDP) (see (Bertsekas and Tsitsiklis, 1996),(Secomandi, 2000)). In such a way the original functional optimization problem is reduced to a nonlinear programming one consisting in selecting the optimal values for the "free" parameters of the neural approximators (see also (Zoppoli *et al.*, 2002) and the references therein). Thanks to the powerful approximating properties of the latter, the original problem can be approximated to any desired degree of accuracy. All the proofs are omitted for the sake of brevity.

2. STATEMENT OF THE PROBLEM

Let us consider a *Decision Maker* (DM) moving on an undirected graph $G = (V, E)$ where

- $V = \{1, 2, ..., N\}$ is the set of nodes;
- E is the set of undirected links (i, j) connecting the nodes i and j; let $M = |E|$;
- c_{ij} = length of link (i, j);
- $E_s \subseteq E$ is the set of stochastic links, whose lengths can take on the values $c_{i,j} = C_{i,j}$ or $c_{i,j} = \infty$, where $0 < C_{i,j} < \infty$; let $M_s = |E_s|$.

For each $(i, j) \in E_s$, let us define a random variable $\theta_{i,j}$ that represents the existence of the corresponding link, and can take on two values

$$\theta_{i,j} = \begin{cases} 1 \text{ if } c_{i,j} = C_{i,j} \\ 0 \text{ if } c_{i,j} = \infty\,. \end{cases} \qquad (1)$$

Moreover, let $\theta = \mathrm{col}[\theta_{i,j}, (i, j) \in E_s]$ and $p(\theta)$ the related a-priori probability mass function. For the sake of simplicity, a two value probability mass function is considered. However, the technique proposed in the following is suitable to be applied in the case of more complex discrete probability density functions.

Before starting, the DM perfectly knows the topology of the graph, but it has only a probabilistic knowledge on the subset E_s of links. When the DM visits a node n for the first time, the lengths of every link departing from n become known. The DM gathers information about any link (i, j) not departing from n according to a given noisy measurement equation $\tilde{g}^{i,j}(n, \theta, \xi)$, where ξ is a random vector noise affecting the measure. To summarize this, let us introduce the following measurement function

$$g^{i,j}(n, \theta, \xi) = \begin{cases} 1 & \text{if } (\ n = i \text{ or } n = j\ ) \text{ and } \theta_{i,j} = 1 \\ 0 & \text{if } (\ n = i \text{ or } n = j\ ) \text{ and } \theta_{i,j} = 0 \\ \tilde{g}^{i,j}(n, \theta, \xi) & \text{otherwise}\,. \end{cases}$$

Moreover, let $g(n, \theta, \xi) = \mathrm{col}[g^{i,j}(n, \theta, \xi), (i, j) \in E_s]$. At time t, when the DM is at node n_t, the measurement vector y_t is obtained as

$$y_t = g(n_t, \theta, \xi_t) \qquad (2)$$

where ξ_t is a general i.i.d. stochastic process. It is worth noting that the above proposed measurement equation can effectively model many real exploration problems. As an example let think to a robotic exploration problem, in which the DM can acquire "perfect" information only on the neighboring portion of terrain, but its sensors are affected by an uncertainty that increases proportionally with the distance. Another example is the exploration of an "unknown" telecommunication network, where the information about links not departing from the current node can be corrupted by noise. Without loss of generality, the links adjacent to the starting node n_0 will be assumed to be a-priori known.

We shall assume the DM to have a perfect memory, then the *information vector* can be defined as

$$I_t \triangleq \mathrm{col}\,[y_0, y_1, \ldots, y_t]\,, \quad t = 0, 1, \ldots\,.$$

Let $p(\theta|I)$ be the conditional probability mass function of the stochastic vector θ when the information vector I has been acquired. Such a mass function can be initialized with the a-priori mass function $p(\theta)$ $\theta_{i,j} \in E_s$ and can be updated by the Bayes law.

It will be useful to describe the movement of the DM on the graph by introducing the following elementary discrete-time dynamic system:

$$\begin{aligned} n_{t+1} &= u_t, \quad t = 0, 1, \ldots, \\ u_t &\in U(n_t, I_t) \end{aligned} \qquad (3)$$

where $n_0 \in V$ and

$$U(i, I) = \Big\{ j \in V : \exists\,(i, j) \in E \setminus E_s \text{ or } \exists\,(i, j) \in E_s,\, p(\theta_{i,j} = 1|I) = 1 \Big\}$$

is the set of "known" neighboring nodes. In particular, $U(n_t, I_t)$ corresponds to the set of all the nodes connected to n_t by a finite-length link.

At any stage $t = 0, 1, \ldots$, the DM makes its decision on the basis of the current node n_t and of the information vector I_t, that is, by the *control function*

$$u_t = \gamma(n_t, I_t), \quad t = 0, 1, \ldots\,. \qquad (4)$$

We have described how the DM moves and acquires information on the graph. Let us now define the objective the DM must achieve. Informally speaking, the goal of the DM is to gather all the possible information about the graph with the minimum path length. In order to formalize this, in the following we shall use the concepts of *information* and *entropy* as given by Shannon (Shannon, 1948). The entropy related to the random vector θ when the information vector I has been collected is defined as

$$H(\theta|I) = - \sum_{\bar{\theta} \in \{0,1\}^{M_s}} p(\theta = \bar{\theta}|I) \log p(\theta = \bar{\theta}|I)\,.$$

Let

$$\mathcal{I}(I', I'') = -\Delta H(I', I'') = H(\theta|I') - H(\theta|I'').$$

$\mathcal{I}(I', I'')$ represents the difference between the "quantity of knowledge" on the stochastic parameters related to two different information vectors I'' and I'. In particular, $\mathcal{I}(I_{t_1}, I_{t_2})$ is the information gain acquired by the DM in the time interval $[t_1, t_2]$. At time t, given an information vector I_t and the current position n_t, the exploration task can be considered "completed" if, in the future, the DM cannot gather an information greater than a given ϵ, i.e.,

$$\lim_{\bar{t}\to\infty} \max_{I_{\bar{t}}} \mathcal{I}(I_t, I_{t+\bar{t}}) \leq \epsilon \qquad (5)$$

subject to (2) and (3). The scalar ϵ represents the desired accuracy. The existence of the maximum is guaranteed by the boundedness of the information gain, since $\mathcal{I}(I', I'') \leq H(\theta) \leq |Es|, \quad \forall I', I''$.

Let us denote as T the a-priori unknown time at which the DM completes its exporation task by satisfying the constraint (5), then the process cost can be defined as

$$J \triangleq \sum_{t=0}^{T-1} h(n_t, u_t) = \sum_{t=0}^{T-1} c_{n_t, u_t} \,. \qquad (6)$$

We can now state the Exploration problem in the form of a usual stochastic optimal control problem.

Problem SGEP (Stochastic Graph Exploration Problem) *Find the optimal control function* γ° *generating* $u_0^\circ = \gamma^\circ(n_0, I_0), \ldots, u_{T-1}^\circ = \gamma^\circ(n_{T-1}, I_{T-1})$ *that minimize the expected value of the cost* J *subject to the constraints (3) and (5).* T *has to be viewed as an a-priori unknown variable.* □

3. APPLYING DYNAMIC PROGRAMMING

The Stochastic Graph Exploration Problem described in the previous section can be ideally solved by means of DP. Let us remark that the dimension of the information vector $I_t, \ t = 0, 1, \ldots$ grows with time. To avoid this heavy drawback, we shall adopt an equivalent formulation for the control function making use of a *sufficient statistic* and, in particular, of the conditional probability mass function $p(\theta|I_t)$ (see e.g. (Bertsekas, 2001)).

For the sake of simplicity and without loss of generality, let us suppose that the variables $\theta_{i,j}, \ (i,j) \in E_s$ remain uncorrelated. Then the sufficient statistic can be concisely represented by means of the M_s-dimensional vector

$$p_t \triangleq \text{col}\left[p_t^{i,j}, \ (i,j) \in E_s\right]$$

where $p_t^{i,j} \triangleq p(\theta_{i,j} = 1|I_t), \ (i,j) \in E_s$ for $t = 1, 2, \ldots$ and $p_0^{i,j}$ are the a-priori probabilities $p(\theta_{i,j} = 1)$. Moreover, let us denote by

$$p_{t+1} = P^+(p_t, y_{t+1})$$

the application of the Bayes formula at stage $t = 1, 2, \ldots$.

In the general case, when the measurement channel is affected by noise, the vectors $p_t, \ t = 0, 1, \ldots$ belong to a continuous space $[0,1]^{M_s}$.

Let us define the "augmented state" corresponding to a node $n \in V$ and a sufficient statistic p as

$$x \triangleq \text{col}(n, p) \,.$$

Then, with a little abuse of notation, the control function (4) can be substituted by $u_t = \gamma(x_t)$. Similarly we shall write $U(x)$ instead of $U(n, I)$. Moreover, let us define as $\hat{S} \triangleq V \times [0,1]^{M_s}$ the set of all the possible augmented states. Of course, since the DM has a perfect knowledge on the adjacent links, not all the nodes $n \in V$ are provided with all the free values of $p \in [0,1]^{M_s}$. Then, in general, only the set $S \subset \hat{S}$ of feasible augmented state has to be considered.

Application of DP yields

$$\begin{aligned} J^{(k+1)}(x) = \min_{u \in U(n)} \Big\{ & h(x,u) \\ & + \mathop{\mathrm{E}}_{\theta,\xi} J^{(k)}\left[\text{col}\left(u, P^+(p, g(u,\theta,\xi))\right)\right] \Big\}, \\ & \qquad\qquad k = 0, 1, \ldots \\ J^{(0)}(x) = \begin{cases} \tilde{J}(x), & \forall x \in S \setminus S_f \\ 0, & \forall x \in S_f \end{cases} & \qquad (7) \end{aligned}$$

where $S_f \triangleq \{x : \mathcal{I}_{\max}(x) \leq \epsilon\}$ and, for a generic augmented state $x = \text{col}(n,p)$, $\mathcal{I}_{\max}(x)$ is the maximum information gain (see (5)), achievable by the DM when $n_t = n$ and $p_t = p$. $\tilde{J}(x)$ are some upper bounds on the optimal costs $J^\circ(x)$.

In order to solve such a problem, a possibility consists in resorting to approximating techniques such as extended Ritz method or NDP (see for example (Zoppoli *et al.*, 2002; Bertsekas, 2001)). In the following of the paper, we shall apply NDP to the solution of the graph exploration problem.

For the sake of simplicity, we shall consider the case where the DM's "vision" is restricted to the adjacent links. In such a simplified framework the measurement equations can be written as

$$y^{i,j} = \begin{cases} \ \ 0 \text{ if } (n = i \text{ or } n = j) \text{ and } c_{i,j} = C_{i,j} \\ \ \ 1 \text{ if } (n = i \text{ or } n = j) \text{ and } c_{i,j} = \infty \\ -1 \text{ otherwise}\,. \end{cases}$$

where by -1 we mean that the DM acquires no information on the parameter, i.e., the measure is uncorrelated with the parameter. In this particular case the vector of measurement noises ξ_t makes no sense ($y_t = g(n_t, \theta)$), and will disappear from now on. Each component of the function P^+ can be written as

$$p_{t+1}^{i,j} = \begin{cases} 0 & \text{if } (n_{t+1} = i \text{ or } n_{t+1} = j) \text{ and } c_{i,j} = C_{i,j} \\ 1 & \text{if } (n_{t+1} = i \text{ or } n_{t+1} = j) \text{ and } c_{i,j} = \infty \\ p_t^{i,j} & \text{otherwise} \end{cases}$$

and each probability $p_t^{i,j}$, $(i,j) \in E_s$ can take on only one of the three values 0, 1, and $p_0^{i,j}$. Hence, the augmented state space S turns out to be a discrete set with cardinality $|S| < N3^{M_S}$. As a consequence it is possible to solve the graph exploration problem by means of exact DP and the recursive algorithm (7) yields the optimal control function $u^\circ = \gamma^\circ(x)$ for any state $x \in S$ in a finite number of iterations. Note that, in this case, the maximum information gain $\mathcal{I}_{\max}(x)$ achievable by the DM in a state $x = \text{col}(n, p)$ can be easily calculated by means of a simple polynomial-time algorithm.

Unfortunately, even in such a simplified framework solving Problem SGEP via the DP algorithm (7) may require an unacceptable computational time, unless very small instances of the problem are involved. In fact, the number of augmentes states is of order $O\left(N3^{M_s}\right)$. Then we may incur the "curse of dimensionality" when the number M_s of stochastic links increases.

In order to mitigate this drawback, in the next sections we shall adopt an approximate technique. Unfortunately, this may lead to a suboptimal control function which is not *proper* (a control function γ is said to be *proper* if it drives the DM from any $x \in S$ to S_f in a finite number of steps). In order to overcome this obstacle, we shall give an equivalent reformulation of problem SGEP for which all the possible policies are proper.

4. AN ALTERNATIVE FORMULATION

Following the lines of (Baglietto *et al.*, 2003*a*), let us define, for a state $x = \text{col}(n, p)$, the set of "frontier nodes" $\tilde{\mathcal{F}}(x)$ as the union all the nodes adjacent to at least one stochastic link with unknown length, i.e.,

$$\tilde{\mathcal{F}}(x) \triangleq \left\{ j \in V \,:\, \exists (j,k) \in E_s, p^{j,k} = p_0^{j,k} \right\} .$$

For any $x = (n, p)$, we shall denote as $\mathcal{F}(x)$ the set of frontier nodes $f \in \tilde{\mathcal{F}}(x)$ such that the shortest path $sp(x, f)$ driving from node n to node f through deterministic links (i.e, on the deterministic graph $(V, E \setminus \{(i,j) \in E_s : p^{i,j} < 1\})$) does not cross any other frontier node $f' \in \tilde{\mathcal{F}}(x)$, $f' \neq f$.

Denote by $U'(x) = \{sp(x,f) : f \in \mathcal{F}(x)\}$ the set of the shortest paths driving from x to any node in $\mathcal{F}(x)$. This set defines the admissible control actions associated to a new Problem which will be called SGEP$'$. In this framework, at any stage $t = 0, 1, \ldots$ the DM chooses the next frontier node to visit or, equivalently, the path in the graph on the basis of the state x, i.e., $u' = \gamma'(x)$ and $u' \in U'(x)$.

In the following, we shall denote by $f(u')$ the frontier node associated to a path u', and by $h'(x, u')$ its deterministic length. Consequently, a new discrete-time dynamic system can be introduced (the integer T' is a stochastic variable):

$$n_{\tau+1} = f(u'_\tau), \quad \tau = 0, 1, \ldots, T' - 1 \qquad (8)$$

$$x_{T'} \in S_f \qquad (9)$$

$$u'_\tau \in U'(x_\tau) \ . \qquad (10)$$

Since a path $u'_\tau \in U'(x_t)$ does not cross any frontier node but $f(u'_\tau)$, all the state transitions associated to such a path are deterministic except for the last one which depends on the realization of $y(f(u'_\tau), \theta)$. Hence

$$p_{\tau+1} = P^+[p_\tau, g(f(u'_\tau), \theta)] \ .$$

We have now all the elements necessary to define an alternative formulation of the cost (6)

$$J' = \sum_{\tau=0}^{T'-1} h'\left(x_\tau, u'_\tau\right)$$

and to state Problem SGEP$'$.

Problem SGEP$'$ *For every $x \in S \setminus S_f$, find the optimal control function γ'° that minimizes the expected value of the cost J' subject to (8)-(10) (T' is the a-priori unknown time at which the DM reaches S_f).* □

According to the definition of the set of admissible controls $U'(x)$, at every time step $\tau = 0, 1, \ldots$ the DM visits a node adjacent to at least one unknown stochastic link. Hence, after at most $M_s + 1$ steps, the DM has a perfect knowoledge of the graph. Then the following proposition holds.

Proposition 1. All the policies for Problem SGEP$'$ reach S_f in at most $M_s + 1$ steps.

Given a control law γ' for Problem SGEP$'$, it is always possible to define an induced control law $\bar{\gamma}$ for Problem SGEP, by choosing, for every state x, $\bar{\gamma}(x)$ as the first node of the path $\gamma'(x)$. Here and in the following given a control law γ for Problem SGEP, we shall denote as $J^\gamma(x)$ the expected value of the cost associated with such a control law, i.e.,

$$J^\gamma(x) \triangleq \mathop{\mathrm{E}}_{\theta} \left\{ \sum_{t=0}^{T-1} h(x_t, \gamma(x_t)) \right\}$$

under the constraints (2) and (3) with $x_0 = x$ and $u_t = \gamma(x_t)$. Clearly given a control law γ' for Problem SGEP′ and the corresponding induced control law $\bar{\gamma}$, we have $J'^{\gamma'}(x) = J^{\bar{\gamma}}(x)$.

The following theorem enlightens the relation between an optimal control law for Problem SGEP′ and the induced control law for Problem SGEP.

Theorem 1. Suppose that γ'° is an optimal control law for Problem SGEP′, and let $\bar{\gamma}^{\circ}$ be the control law for Problem SGEP derived from γ'°. Then $\bar{\gamma}^{\circ}$ is an optimal control law for Prolem SGEP.

In the following we shall consider Problem SGEP′, since, in the light of Theorem 1, it turns out to be equivalent to Problem SGEP. While, on one hand, this choice requires a little computational overhead, on the other hand, we can look for an approximate solution without having to check if it is proper (see Proposition 1).

5. APPROXIMATE VALUE ITERATION

Problem SGEP′ can be solved exactly by means of a DP algorithm similar to (7). More specifically, the following result can be claimed.

Proposition 2. The DP algorithm yields the optimal control function $u'^{\circ} = \gamma'^{\circ}(x)$ for any state $x \in S \setminus S_f$ in at most M_s iterations.

However, as stated previously, since the number of the states $S \setminus S_f$ grows exponentially with the number M_s of stochastic links, for complex instances of the graph it is not possible to find the optimal cost function $J^{\circ}(x)$ in a reasonable time by using the "exact" algorithm. Hence, we shall resort to an approximation technique that consists in assigning a given structure to the cost-to-go function. In such a structure, a certain number of "free" parameters have to be determined in order to approximate as well as possible the optimal cost-to-go function $J^{\circ}(x)$. Following (Zoppoli *et al.*, 2002), we choose as fixed-structure functions the so called "one-hidden-layer" (OHL) networks. This means that, for each node, the approximate cost-to-go takes on the form

$$\widehat{J}_n(p, w_n) = \sum_{i=1}^{\nu_n} c_{n,i}\, \varphi(p, w_{n,i}),\ n = 1, 2, \ldots, N$$

where ν_n is the number of parametrized basis functions of the n-th approximator and $w_n \triangleq \mathrm{col}(w_{n,1}, c_{n,1}, w_{n,2}, c_{n,2}, \ldots, w_{n,\nu_n}, c_{n,\nu_n})$ is the vector of "free" parameters to be tuned. Furthermore, if we define the vector of all the parameters as $w \triangleq \mathrm{col}\ (w_0, w_1, \ldots w_{N-1})$, then we can write the approximate cost-to-go function $\widehat{J}(\cdot)$ as

$$\begin{aligned} &\widehat{J}(x, w) = \widehat{J}_n(p, w_n),\ \forall x = \mathrm{col}(n, p),\ x \in S \setminus S_f \\ &\widehat{J}(x, w) = 0, \qquad\qquad \forall x \in S_f\,. \end{aligned}$$

Among various possible parametrized basis functions φ, we choose sigmoidal functions $\sigma(\bar{w}_{n,i} \cdot p + w_{n,0i})$. Then the approximators $\widehat{J}_n$ are given by OHL neural networks. As to the capability of OHL neural networks to approximate the optimal solutions the reader is referred to (Baglietto *et al.*, 2003*a*).

Clearly, following the guideline of the formulation of Problem SGEP′, it is possible to associate a control function to any given cost-to-go function $\widehat{J}$ by means of the DP operator. By construction, the resulting control law turns out to be proper independently of the values of the cost-to-go $\widehat{J}(x)$, $x \in S \setminus S_f$.

We are now able to formulate a mathematical programming problem that approximates the original functional Problem SGEP′ to any degree of accuracy (the reader interested in the approximation of functional stochastic optimization problems by approximating parametrized schemes is referred to (Zoppoli *et al.*, 2002)).

Problem SGEP$_w$ *Find the optimal vector* $w^{\circ} = \mathrm{col}\left(w_0^{\circ}, w_1^{\circ}, \ldots, w_{N-1}^{\circ}\right)$ *such that the control function* $\widehat{\gamma}'$ *associated to the approximate cost-to-go functions* $\widehat{J}_n(I, w_n^{\circ})$, $n = 1, 2 \ldots, N$, *minimizes* $\sum_{x \in \{S \setminus S_f\}} J'^{\widehat{\gamma}'}(x)$. □

We now describe in some detail the "approximate value iteration" algorithm that can be used to determine w°. Such an algorithm is similar to the incremental approximate value iteration algorithm described in (Bertsekas and Tsitsiklis, 1996) but, in this case, N different neural networks are trained at the same time.

Algorithm 1.

1. Choose randomly the initial weight vectors $w_n^{(0)}$, $n = 1, 2, \ldots, N$; set $k = 0$;

2. choose randomly a state $x^{(k)} \triangleq \mathrm{col}(n, p) \in S \setminus S_f$;

3. make one step of the value iteration algorithm in the state $x^{(k)}$:

$$\begin{aligned} \bar{J}\left(x^{(k)}\right) = \min_{u' \in U'(x^{(k)})} &\Big\{ h'\left(x^{(k)}, u'\right) \\ &+ \mathop{\mathrm{E}}_{\theta}\left[\widehat{J}_{f(u')}\left(P^{+}\left(p, g(f(u'), \theta)\right), w_{f(u')}^{(k)}\right)\right]\Big\}; \end{aligned}$$

4. update the weight vectors of the N neural networks according to

$$\begin{aligned} w_n^{(k+1)} &= w_n^{(k)} \\ &- \frac{c_1}{c_2 + k} \nabla_{\boldsymbol{w}_n} \left\{ \left[\ \widehat{J}_n\left(p, w_n^{(k)}\right) - \bar{J}\left(x^{(k)}\right)\ \right]^2 \right\} \end{aligned}$$

and $w_i^{(k+1)} = w_i^{(k)}, \quad \forall i \neq n,\, n \in V$;

5. if

$$\sum_{\boldsymbol{x}\in S\setminus S_f}\left[\,\widehat{J}\big(x,w^{(k+1)}\big)-\widehat{J}\big(x,w^{(k)}\big)\,\right]^2>\delta\,.$$

then set $k=k+1$ and return to step 2.

Note that, since the number of feasible states $|S\setminus S_f|$ grows exponentially with the number of stochastic links, for complex instances of the graph G the computation of the summation in step 5 may require too much time. Hence, such a summation can be computed over a validation set $VS^{(k)}$, composed by a given number $\alpha \ll |S\setminus S_f|$ of admissible states.

6. NUMERICAL RESULTS

Let us consider a graph with 9 nodes and 6 stochastic links (see Fig. 1) and let us suppose that the DM's vision is restricted to the adjacent links. Moreover let us suppose that $\epsilon=0$, i.e., the goal of the DM is to explore all the reachable stochastic links. The a-priori probabilities $p_0^{i,j}$ have been chosen equal to $\frac{1}{2}$.

In this case, we have $|\hat{S}| = 6561$ and, after the exclusion of all the unfeasible states, $|S| = 4050$. Given the simplicity of the graph and the relatively small number of states, we can apply the exact DP algorithm (which ends in 6 iterations) to find the optimal control function $\gamma'^{\circ}(x)$ and the optimal cost to go $J'^{\circ}(x)$, $\forall x\in S\setminus S_f$.

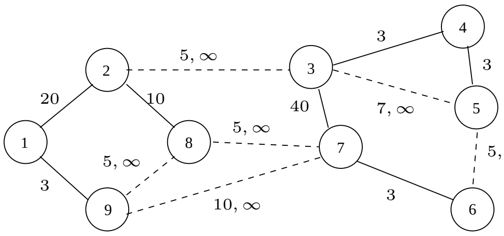

Fig. 1. A simple stochastic graph.

The optimal solution has been compared with the approximate solution obtained by following the approach of Section 5. For each node a OHL neural network with 10 hyperbolic tangent activation functions has been used. The neural networks have been trained by means of Algorithm 1. Let $\widehat{J}$ be the approximate cost function after the training process and let $\hat{\gamma}'$ be the proper control function derived from the approximate cost function $\widehat{J}$ by means of the DP operator. For the sake of comparison we considered the percentage error between the approximate cost-to-go function $\widehat{J}$ and the optimal one J°, that is,

$$PE(x)\overset{\triangle}{=}\big|\widehat{J}(x,w)-J^{\circ}(x)\big|/J^{\circ}(x),\quad \forall x\in S\setminus S_f$$

and the percentage error between the cost function $J^{\hat{\gamma}'}$ and the optimal one J'°, defined as

$$PE'(x)\overset{\triangle}{=}\big|J'^{\hat{\gamma}'}(x)-J^{\circ}(x)\big|\;J^{\circ}(x),\quad \forall x\in S\setminus S_f\,.$$

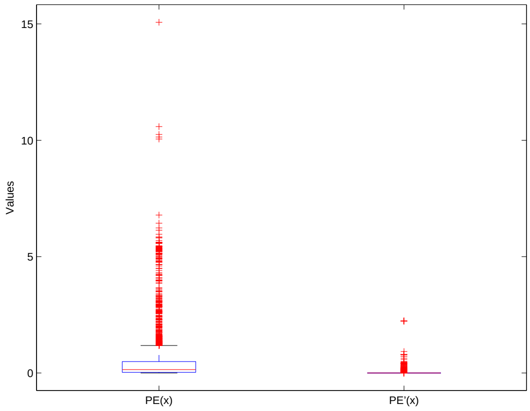

Fig. 2. Box plots of the percentage errors PE and PE'.

As can be seen from Fig. 2, even if the approximate function $\widehat{J}$ does not represent a very good approximation of the optimal cost function J°, the costs-to-go of the proper policy $\hat{\gamma}'$, derived from $\widehat{J}$, turn out to be very close to the optimal ones.

The choice of a very simple graph is motivated by the possibility of a direct comparison with the exact solution (see Fig. 2). However, the proposed approach for the approximate solution of Problem SGEP is well-suited to being applied to more complex graphs, for which the exact solution cannot be computed due to computational issues. A simulation analysis in this case has been performed but it is omitted for the sake of brevity.

REFERENCES

Baglietto, M., G. Battistelli, F. Vitali and R. Zoppoli (2003*a*). Shortest path problems on stochastic graphs: a neuro dynamic programming approach. In: *42nd IEEE Conference on Decision and Control*. Maui, Hawaii. pp. 6187–6193.

Baglietto, M., L. Scardovi and R. Zoppoli (2004). Active identification of unknown systems: an information theoretic approach. In: *42nd IEEE American Control Conference*. Boston. to appear.

Baglietto, M., M. Paolucci, L. Scardovi and R. Zoppoli (2003*b*). Entropy-based environment exploration and stochastic optimal control. In: *42nd IEEE Conference on Decision and Control*. Maui, Hawaii. pp. 2938–2941.

Bertsekas, D. P. (2001). *Dynamic Programming and Optimal Control*. Athena Scientific.

Bertsekas, D. P. and J. N. Tsitsiklis (1996). *Neuro-Dynamic Programming*. Athena Scientific.

Deng, X. and C. Papadimitriou (1999). Exploring an unknown graph. *Journal of Graph Theory* **32**, 265 –297.

Secomandi, N. (2000). Comparing neuro-dynamic programming algorithms for the vehicle routing problem with stochastic demands. *Computers & Operations Research* **27**, 1201–1225.

Shannon, C. (1948). A mathematical theory of communication. *Bell System Technical Journal* **27**, 379–423.

Zoppoli, R., M. Sanguineti and T. Parisini (2002). Approximating networks and the extended ritz method for the solution of functional optimization problems. *Journal of Optimization Theory and Applications* **112**, 403–439.