Reinforcement Learning Adaptive Control and Explicit Criterion Maximization

Tomas Landelius, tc@isy.liu.se
Computer Vision Laboratory, Linköping University
S-581 83 Linköping, Sweden

Abstract

This paper reviews an existing algorithm for adaptive control based on explicit criterion maximization (ECM) and presents an extended version suited for reinforcement learning tasks. Furthermore, assumptions under which the algorithm convergences to a local maxima of a long term utility function are given. Such convergence theorems are very rare for reinforcement learning algorithms working with continuous state and action spaces. A number of similar algorithms, previously suggested to the reinforcement learning community, are briefly surveyed in order to give the presented algorithm a place in the field. The relations between the different algorithms is exemplified by checking their consistency on a simple problem of linear quadratic regulation (LQR).

1 Introduction

It is only recently that the close relationship between reinforcement learning and adaptive optimal control has been recognized [5]. An excellent overview and introduction to these relations is provided in the reference [10]. Much of the material on dynamic programming and the descriptions of the algorithms used in the comparative studies in this article originates from this book.

There are however some differences between reinforcement learning and adaptive optimal control. In optimal control it is assumed that a function describing the immediate or instantaneous system performance is provided, often after a lot of hard work, by the system designer. This is not the view in reinforcement learning, where the system also has to find out how its actions are rewarded instantaneously.

In this paper we will present an algorithm which we have brought from the field of optimal control and extended in order to cope also with the case when the immediate payoff function is unknown to the system [6]. The review of the algorithm is adopted from this reference. Neither is it an algorithm for approximate dynamic programming, even though there are some strong similarities as will be pointed out later, nor does it work off-line as do many other algorithms proposed for maximization of payoff in the long term.

2 System and regulator

In this paper we will suppose that the system to be controlled can be described with a parameterized function in the following way:

$$y(t) = y(\theta, \varphi(t)) + e(t). \tag{1}$$

Here $y(t)$ is the system output at time t, $e(t)$ is a zero mean noise, and $\varphi(t)$ is a regressor consisting of previous inputs and outputs

$$\varphi^T(t) = (-y^T(t-1), \ldots, -y^T(t-n), u^T(t-1), \ldots, u^T(t-m), \tag{2}$$

where $u(t)$ is the system input which is also equal to the regulator output:

$$u(t) = u(w, y(t), \varphi(t)). \tag{3}$$

Hence we assume the structure of the regulator to be given and concentrate on adjusting the parameters w in order to get a good closed loop performance. When we close the loop both the systems input and output will depend on the parameters for the regulator, i.e. we will have $y(t, w)$ and $u(t, w)$.

In the comparisons carried out in section 6, special attention will be put on the case where the system and regulator dynamics can be described with linear difference equations:

$$\begin{aligned} y(t) + a_1 y(t-1) + \ldots + a_n y(t-n) &= b_1 u(t-1) + \ldots + b_m u(t-m) & (4)\\ u(t) + r_1 u(t-1) + \ldots + r_m u(t-m) &= s_1 y(t-1) + \ldots + y_n y(t-n). & (5) \end{aligned}$$

In this case it is also possible to formulate a linear state-space model for the system and the regulator:

$$\begin{array}{rclcl} x(t+1) &=& f(x(t), u(t)) + e(t) &=& Ax(t) + Bu(t) + e(t) \\ y(t) &=& Cx(t) && \\ u(t) &=& h(w(t), x(t)) &=& Wx(t). \end{array} \tag{6}$$

3 The mission

As mentioned in the introduction we are looking for regulator parameters w in order to maximize a measure of success in the long run. It is seldom the case that you explicitly know how the regulator output affect the payoff in the long term. Without such knowledge you need some sort of a model to predict how regulator outputs will change the state of the system to be controlled in order to evaluate the impact of parameter changes on future payoff.

As in optimal control we define a measure of long term success by a functional which is a time integral, beginning with the system being in state $x(t)$, of the instantaneous reward g:

$$V(w, x(t)) = \int_t^\infty g(x(t,w), u(t,w))dt. \tag{7}$$

Since we deal with a time discrete and noisy system it is natural to extend the definition above to:

$$V(w, x(t)) = E\{\sum_{k=t}^{\infty} g(x(k,w), u(k,w))\}. \tag{8}$$

In what follows we will emphasize the case when the regulation leads to a closed loop system with stationary input and output signals, i.e. their mean and covariances will be time invariant. Hence minimizing V in eq. 8 will be equal to minimizing the expected value of the instantaneous reward

$$V(w) = E\{g(x(t,w), u(t,w))\} = E\{g(Cy(t,w), u(t,w))\}. \tag{9}$$

4 Explicit criterion maximization

In this section we will review and extend an on-line algorithm for explicit criterion maximization (ECM) where the closed loop system is stationary [6]. The explicit criterion is hence the utility function in eq. 9. A natural way to adjust the parameter w is to compute the gradient of the utility function with respect to the parameters and then update them in the direction of the gradient. This basic scheme is often referred to as the *MIT rule* in control theory [1]. The following algorithm is known to converge to a local maximum of $E\{g\}$ under some regularity conditions [6]:

$$w(t+1) = w(t) + \gamma(t)H^{-1}(t)\frac{\partial g}{\partial w}(t), \tag{10}$$

where $\gamma(t)$ is a gain sequence and H(t) is a positive definite matrix. The choice of suitable step lengths and approximations of the inverse of the Hessian will not be discussed here, instead the reader is referred to the vast literature on this subject.

Now to the question of estimating the gradient of the instantaneous reward $g(Cy(w,t), u(w,t))$. Differentiation with respect to w results in

$$\frac{\partial g}{\partial w} \propto \frac{\partial g}{\partial y}\frac{\partial y}{\partial w} + \frac{\partial g}{\partial u}\frac{\partial u}{\partial w}. \tag{11}$$

The original formulation of this algorithm was based on the assumption that the derivatives of g with respect to y and u where known. In many cases it is more reasonable to view the instantaneous reward to be produced by the environment and not man made and built into the system. This means that the system has to estimate these derivatives as well. One possibility would be to parameterize the reward and use the *instrumental variable* (IV) approach for closed loop identification suggested in [6].

For the other two derivatives we end up with these expressions:

$$\frac{\partial y}{\partial w} = \frac{\partial y}{\partial \varphi}\frac{\partial \varphi}{\partial w} \quad \text{and} \quad \frac{\partial u}{\partial w} = \frac{\partial u}{\partial w} + \frac{\partial u}{\partial y}\frac{\partial y}{\partial w} + \frac{\partial u}{\partial \varphi}\frac{\partial \varphi}{\partial w}. \tag{12}$$

For the moment, assume that $(\partial y/\partial w)(s)$ and $(\partial u/\partial w)(s)$ are known for $s < t$, then also $(\partial\varphi/\partial w)(s)$ is known according to eq. 2. Together with the assumption that the derivative of the system model with respect to the regressor φ and the derivative of the regulator with respect to w are known it turns out that first $\partial y/\partial w$ and from this $\partial u/\partial w$ can be calculated by recursion. Note that these recursive calculations, or filters, are stable if the closed loop is.

In order to fulfill the assumption that $\partial y/\partial\varphi$ is known we need to know the systems dynamics e.g. by assuming that the function $y(\theta, \varphi(t))$ as well as its parameters θ are known. The problem of $y(w,t)$ being unobservable because of the constant adjustment of the regulator parameters w can be solved by using the current parameter values in the recursive filter calculation above [4].

Summing up the parts, the complete algorithm can be described according to the following time ordered sequence, where $\hat{\theta}$ and $\hat{g}$ denote estimates from the system identification procedure:

$$\begin{aligned}
y(t) &= y(\theta, \varphi(t)) + e(t) && (13)\\
u(t) &= u(w(t-1), y(t), \varphi(t)) && (14)\\
\frac{\partial y}{\partial w} &= \frac{\partial y(\hat{\theta}(t), \varphi(t))}{\partial\varphi}\frac{\partial\varphi}{\partial w} && (15)\\
\frac{\partial u}{\partial w} &= \frac{\partial u(w(t-1))}{\partial w} + \frac{\partial u(w(t-1))}{\partial y}\frac{\partial y}{\partial w} + \frac{\partial u(w(t-1))}{\partial\varphi}\frac{\partial\varphi}{\partial w} && (16)\\
w(t) &= w(t-1) + \gamma(t)H^{-1}\left(\frac{\partial\hat{g}}{\partial y}\frac{\partial y}{\partial w} + \frac{\partial\hat{g}}{\partial u}\frac{\partial u}{\partial w}\right). && (17)
\end{aligned}$$

The converge properties of this algorithm has been investigated [6] and it turns out that convergence to a local maximum of the criterion in eq. 8 can be proved given that the estimates $\hat{\theta}$ and $\hat{g}$ converge to their true values with probability 1 and assuming boundedness of the noise, the gradient of the utility function, and the regressor and regulator parameters. It is also necessary that the regulator defines a closed loop system with all its poles strictly inside the unit circle. Again the closed loop (IV) approach proposed in [6] can be used to identify the system with probability 1 as $t \to \infty$.

5 Relations to other algorithms

This section will relate the algorithm outlined in the previous section to some other similar classes of algorithms for reinforcement learning.

Backpropagation of utility This class of algorithms assumes that an exact model of the system is at hand. In order to update the regulator parameters the derivatives of the utility function with respect to the state is calculated an used for a steepest ascent search. The difference between most of the algorithms in this class and the algorithm from the previous section is that the search is done off-line, using *backpropagation-through-time* (BTT) [3], or some other method for backward integration of the derivative. Before we calculate the derivative we rewrite eq. 8 on a recursive form:

$$V(w, x(t)) = E\{g(x(t), u(t,w)) + V(f(x(t), u(t,w)))\}. \tag{18}$$

With this definition of utility the sensitivities now become:

$$\frac{\partial V}{\partial x} = E\{\frac{\partial g}{\partial x} + \frac{\partial g}{\partial u}\frac{\partial u}{\partial x} + \frac{\partial V(x(t+1))}{\partial x(t+1)}\left(\frac{\partial x(t+1)}{\partial x} + \frac{\partial x(t+1)}{\partial u}\frac{\partial u}{\partial x}\right)\}. \tag{19}$$

For a problem with a finite horizon where we do not let N in eq. 8 go to infinity. Then we can use the fact that initial values are know for the states, $x(0)$ and final values are known for $\partial V/\partial x(N)$ and find the sensitivities for time t by integrating backwards. Once the sensitivities are known we proceed and adjust the regulator parameters according to:

$$\begin{aligned} \frac{\partial}{\partial w}V(w,x(t)) &= E\{\frac{\partial}{\partial w}g(x(t),u(w,t)) + \frac{\partial}{\partial w}V(f(x(t),u(w,t)))\} \quad (20) \\ &= E\{\frac{\partial g}{\partial w}\frac{\partial u}{\partial w} + \frac{\partial V}{\partial f}(\frac{\partial f}{\partial u}\frac{\partial f}{\partial w})\}. \end{aligned}$$

Adaptive critics If we want to find the regulator parameters that gives an optimal value to V and the parameters of the system are uncertain there is only one class of algorithms that can give us the answer - dynamic programming algorithms [1]. Note that conditions for the existence of an optimal regulator does not exist. However if one exists the optimal functional V^* and regulator h^* can be found by solving the *Bellman equation*:

$$\begin{aligned} V^*(x(t)) &= \max_{u(t)} E\{g(x(t),u(t)) + V^*(f(x(t),u(t)))\} \quad (21) \\ h^*(x(t)) &= \arg\max_{u(t)} E\{g(x(t),u(t)) + V^*(f(x(t),u(t)))\} \quad (22) \end{aligned}$$

Even though this equation looks simple it is only possible to solve it numerically and only in very simple cases. The Bellman equation tells us that to receive maximal payoff in the long term we should have a regulator that maximizes the expected sum of the instantaneous reward and the expected long term reward accessible from the next state.

An interesting successive procedure for obtaining V^* has been suggested by Howard [3]. He showed that alternating between the following two steps leads to the optimal solution:

1. Find a model of $V(w,x(t))$ for the present parameters w.
2. Update the parameters w so that this estimate of V is maximized.

Adaptive critics is the name of a class of algorithms that try to approximate dynamic programming. A number of different algorithms have been proposed and this comparison will only deal with two types, both of which are coined by Werbos [8]:

HDP heuristic dynamic programming which attempts to model the functional V. TD-methods [2] and Q-learning [7] can be seen as special cases of HDP.

DHP dual heuristic programming tries to model the derivatives of the functional V. Much because the derivative contains information on in which direction to change our parameters there seems to be a number of advantages in favor of this type of algorithms compared to the HDP approach [9].

Modelling V or its gradient with respect to the state variable will not explicitly give us an optimal regulator. Again there are many ways to update the regulator parameters w. Here we will consider the backpropagated adaptive critic (BAC), since it is the one that bear the most resemblances to the ECM algorithm presented in section 4 and because it seems to be the algorithm used most frequently.

The basic idea with BAC is to adapt the regulator so that it produces an output $u(t)$ that maximizes the value of the current model of $V(w, x(t))$, i.e. that performs the second step in Howard's algorithm. Some texts seem to suggest that the value of V in the *next state*, $V(x(t+1))$, should be maximized but this will not lead to the desired result, as will be shown in the next section.

Since the update of w is done at time t we should note that this update will not affect $x(t)$ only $u(w,t)$. Hence, we would like to update our parameters in direction of the gradient which again leads us to eq. 20. This adjustment direction is valid for both HDP and DHP. The difference between them lies in how the derivative $\partial V/\partial f = (\partial V/\partial x)(t+1)$ is estimated. In HDP we have a model of V which can be used to find the derivative with respect to the state vector x by means of backpropagation. In DHP, on the other hand, the derivative itself is the modeled entity so it can be used as an estimate as it is.

6 An LQR example

In this section we compare the update rules governing the HDP, DHP, and ECM algorithms on a simple system in order to give a better insight into what is really going on. We also point out a possible pitfall concerning the BAC algorithm.

In this example the system, regulator, and the instantaneous reward function constitute a *linear quadratic regulation* (LQR) problem:

$$\begin{aligned} x(t+1) &= f(x(t), u(t)) + e(t) = ax(t) + bu(t) + e(t) \quad (23)\\ u(t) &= h(w(t), x(t)) = w(t)x(t) \quad (24)\\ g(x(t), u(t)) &= -qx^2(t) - ru^2(t), \quad (25) \end{aligned}$$

where all constants are scalars and the state $x(t)$ equals the current output $y(t)$ $(r, q \geq 0)$. It is well known that the optimal stationary solution to this problem is given by a regulator with the parameters [1]:

$$w = \frac{-bka}{r + bkb} \quad \text{where} \quad k = (1 - (a + bw)^2)^{-1}(q + rw^2). \quad (26)$$

Now lets turn to the ECM algorithm and check if the resulting regulator equals the optimal one. First we note that since the input and output signals are stationary, their covariances are given by:

$$E\{xx\} = (a + bw)E\{xx\}(a + bw) \quad \text{and} \quad E\{uu\} = wE\{xx\}w. \quad (27)$$

The ECM algorithm converges to a local maximum of the utility function in eq. 8, i.e. when:

$$E\{\frac{\partial g}{\partial w}\} = E\{\frac{\partial g}{\partial x}\frac{\partial x}{\partial w} + \frac{\partial g}{\partial u}\frac{\partial u}{\partial w}\} = 0. \quad (28)$$

The derivatives of the instantaneous reward with respect to inputs and outputs equals $2qx$ and $2ru$ respectively. Inserting this into the equation above yields:

$$qE\{2x\frac{\partial x}{\partial w}\} + rE\{2u\frac{\partial u}{\partial w}\} = 0. \tag{29}$$

The two expectation values can now be identified as derivatives of the covariance matrices in eq. 27:

$$\begin{aligned} E\{x\frac{\partial x}{\partial w}\} &= (1-(a+bw)^2)^{-1}(a+bw)bE\{xx\} = k\frac{(a+bw)b)}{q+rw^2}E\{xx\} && (30)\\ E\{u\frac{\partial u}{\partial w}\} &= wE\{xx\} + w^2E\{x\frac{\partial x}{\partial w}\}. && (31) \end{aligned}$$

Expanding eq. 28 in terms of these derivatives and using the expression for k from eq. 26 results in:

$$\begin{aligned} (q+rw^2)E\{x\frac{\partial x}{\partial w}\} + wrE\{xx\} &= 0 && (32)\\ kb(a+bw)E\{xx\} + wrE\{xx\} &= 0, && (33) \end{aligned}$$

which gives the desired parameters in eq. 26 when solved for w. Note that the system must be persistently excited, i.e. have a covariance greater than zero in order for the solution to exist.

Next, lets look at the consistency of the two algorithm classes HDP and DHP which both share the equation for parameter adjustment:

$$E\{\frac{\partial g}{\partial w}\} = E\{\frac{\partial g}{\partial u}\frac{\partial u}{\partial w} + \frac{\partial V}{\partial f}\frac{\partial f}{\partial u}\frac{\partial u}{\partial w}\} = 0. \tag{34}$$

In difference with the ECM calculations the derivatives in the above equation should be evaluated for constant x in line with Howard's algorithm. Using the fact that the utility function for an LQR problem can be written as $V(x) = xkx$, and that $f(x(t), u(t)) = x(t+1)$, eq. 34 is turned into:

$$\begin{aligned} E\{2rux + \frac{\partial}{\partial x(t+1)}(x(t+1)kx(t+1))\frac{\partial}{\partial u}(ax+bu+e)\} &= 0 && (35)\\ rw^2E\{xx\} + (a+bw)kbE\{xx\} &= 0, && (36) \end{aligned}$$

which also results in the optimal parameters when solved for w. However, adjusting the parameters in order to maximize the utility of the next state will not provide the optimal regulator:

$$E\{\frac{\partial V(x(t+1))}{\partial w}\} = E\{\frac{\partial V(x(t+1))}{\partial x(t+1)}\frac{\partial x(t+1)}{\partial u}\frac{\partial u}{\partial w}\} = 2(a+bw)kbE\{xx\}. \tag{37}$$

This seems to be an error made in some presentations of e.g. the BAC algorithm. It is only true if we restrict ourselves to the case where the instantaneous reward is a function of the state only, i.e. $g(x, u) = g(x)$.

7 Conclusions

We have presented an extended version of an algorithm for adaptive control based on explicit criterion maximization. In order to fit into the category of reinforcement learning algorithms the assumption that the function providing the instantaneous reward is known was dropped. Instead we suggest that this function is parameterized and identified using the same machinery as is used for identification of the dynamical system.

Furthermore the algorithm convergences to a local maxima of the utility function under some assumptions about the system and its identification. Such convergence theorems are very rare for reinforcement learning algorithms working with continuous state and actions.

A number of similar algorithms were briefly surveyed in order to place the presented algorithm among the ones previously suggested to the reinforcement learning community. To shed some more light over the relations between the different algorithms their consistency was checked on a very simple problem where both the system and the regulator where linear and the instantaneous payoff was given by a quadratic function.

To conclude it seems that work made in the field of adaptive optimal control may contribute with fresh ideas on new algorithms for reinforcement learning as well as clues on how to establish, or at least investigate, the convergence properties of previously proposed algorithms in the field of reinforcement learning.

References

[1] K. J. Åström and B. Wittenmark. *Adaptive Control*. Addison-Wesley publishing comp., 1989.

[2] A. G. Barto, R. S. Sutton, and C. W. Anderson. Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Trans. on Systems, Man, and Cybernetics*, SMC-13(8):834–846, 1983.

[3] R. A. Howard. *Dynamic programming and Markov processes*. John Wiley & Sons Inc., 1960.

[4] L. Ljung. Analysis of a general recursive prediction erroe identification algorithm. *Automatica*, 17(1):89–99, 1981.

[5] R. S. Sutton, A. G. Barto, and R. J. Williams. Reinforcement learning is direct adaptive optimal control. In *Proc. of the American Control Conf.*, pages 2143–2146, 1991.

[6] E. Trulsson and L. Ljung. Adaptive control based on explicit criterion minimization. *Automatica*, 21:385–399, 1985.

[7] C. Watkins. *Learning from delayed rewards*. PhD thesis, Cambridge University, 1989.

[8] P. Werbos. *Neural Networks for Control*, chapter A menu of designs for reinforcement learning over time. MIT Press, Cambridge, MA, 1990.

[9] P. Werbos. *Handbook of Intelligent Control*, chapter Approximate dynamic programming for real-time control and neural modelling. Van Nostrand Reinhold, 1992.

[10] D. A. White and D. A. Sofge. *Handbook of intelligent control: fuzzy, neural, and adaptive approaches*. New York Van Nostrand Reinhold cop., 1992.