Foreword
ADP: The Key Direction for Future Research in Intelligent Control and Understanding Brain Intelligence

IN THE future, we may recognize that trying to build or understand intelligent systems without exploiting the Bellman equation is like trying to build hardware without knowing Maxwell's laws. There are times when proper understanding and use of one key equation is the key bridge that makes it possible to connect valid large global goals to the world of concrete mathematical reality, i.e., working designs and valid models. New fundamental mathematics is also the key to creating a unification of understanding across disciplines, such as engineering, psychology, neuroscience, and even social science.

It is great pleasure to introduce this special issue of the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS, the journal that was, in a way, the birthplace of the modern adaptive dynamic programming (ADP) community, as I will discuss.

Back in the early 1960s, Minsky [1] proposed that we could reverse engineer the intelligence of the brain by building general-purpose "reinforcement learning systems (RLSs)." He defined an RLS as a kind of universal black box.

At each time t, the RLS inputs a vector of observed variables $\mathbf{X}(t)$, outputs a vector of decisions or controls $\mathbf{u}(t)$, and receives a global reward or punishment signal, which I would denote as $U(t)$. In artificial intelligence, it is more common to refer to $U(t)$ as a "reward" variable r because of the strong connections between reinforcement learning and animal psychology [2]. In the ADP approach, however, we are interested in building powerful intelligent systems, which can address well-defined mathematical tasks. We are focusing on the goal of maximizing the expected future value of a "utility function"—a concept that was made rigorous by the work of Von Neumann and Morgenstern [3], which led to Bellman's later discovery of dynamic programming (DP). Previous work on maximizing utility functions, by Hamilton and Jacobi and Lagrange *et al.*, focused on the case of *deterministic systems*; von Neumann and Bellman achieved the great breakthrough of developing more general mathematics for the *stochastic* case.

The Bellman equation is *the* fundamental equation for defining or deriving the optimal strategy or policy of action, in the general case, for nonlinear stochastic dynamical systems. [4].

Minsky was very disappointed when he found that an early "common sense" version of reinforcement learning could not simply handle as much complexity as ordinary DP, let alone the complexity that the brain can handle. In 1968, I was first to suggest that we could overcome this problem by developing *adaptive approximations* to the Bellman equation as a new way to build RLSs [5]. In a sense, this was the birth or initial formulation of ADP; but in 1968, it was only an idea. Even today, the development of new ADP designs to handle ever more complexity without losing their generality is the number one challenge to research [6].

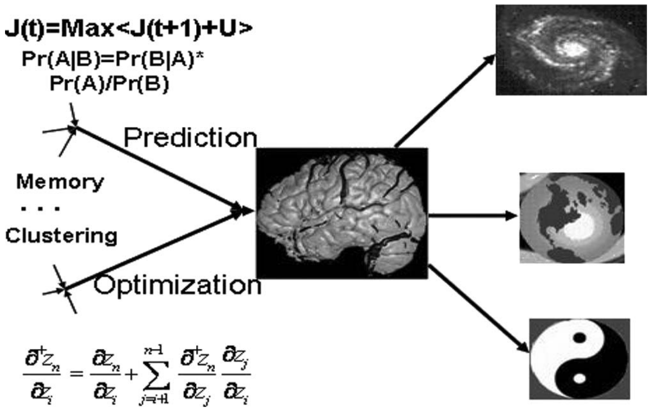
J(t)=Max<J(t+1)+U>
Pr(A|B)=Pr(B|A)*
Pr(A)/Pr(B)
Prediction
Memory
. . .
Clustering
Optimization
$\frac{\partial^+ z_n}{\partial z_i} = \frac{\partial z_n}{\partial z_i} + \sum_{j=i+1}^{n-1} \frac{\partial^+ z_n}{\partial z_j}\frac{\partial z_j}{\partial z_i}$

Fig. 1. Important future directions for research, of interest in NSF funding: 1) developing and combining adaptive brainlike optimal control and prediction and 2) addressing key challenges in engineering for sustainability on Earth, space development, and human potential. (Top left) Schematic of Bellman equation.

In 1971, for my Harvard Ph.D. thesis work [7]–[9], I proposed the first consistent ADP design illustrated in Fig. 1.

To overcome the curse of dimensionality, I proposed that we do what statisticians had been doing for centuries in learning approximate statistical models of the world: develop and train a *parameterized model* of the Bellman J function (the "value function"). In order to converge to the correct value function in the general case, we need to use some kind of universal nonlinear function approximator to serve as the "Critic." Based on the theorems of Barron [10], we know that neural networks can fill that role with less error than more traditional approximators, *when there are many variables in the system.*

How could we implement such a system? Where are the equations to fill in the boxes? From the viewpoint of adaptive control, I proposed that we adapt model and action networks based on exactly the same derivatives that Narendra proposed in his neural network design for indirect adaptive control (IAC) [11]. What I proposed for Fig. 1 was the same as his IAC,

Digital Object Identifier 10.1109/TSMCB.2008.924139

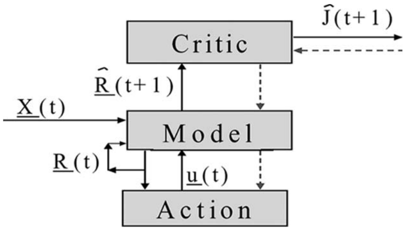

Fig. 2. First ADP design. The dashed lines represent calculations of ordered derivatives, or "backpropagation," in order to calculate the gradient of J with respect to control variables and parameters at *minimum computational cost*. **R** represents an estimated "belief state," which emerges as a byproduct of the system identification component or "Model."

except that I proposed the use of the "Critic" network in place of the square position error used by Narendra. To adapt the Critic network, I proposed the use of a method that I called "heuristic dynamic programming" (HDP), which was later renamed as the "TD(0)" temporal method by Sutton [12]. By replacing the (arbitrary) square error measure with a Critic, we can converge toward an optimal policy and also avoid the large transient errors and common modes of instability that are found when IAC is applied to complicated types of plants. For details, the reader is urged to obtain the two key handbooks from National Science Foundation (NSF) workshops on intelligent control [13] and ADP [14]. Those workshops received support from many parts of NSF, and the books and web site contain pointers to NSF funding interests in that area.

Lewis *et al.* have made tremendous progress in extending HDP and related methods to strengthen the guarantees of stability, similar to the progress that Narendra made decades ago in stabilizing ordinary adaptive control. There is more to be done along those lines, of course, and that is another important direction for future research. Lewis's leadership in this area has become essential to the field.

However, even in 1971, I realized that HDP still had certain problems in scaling up to fast learning in large environments. Thus, I proposed two extended methods for training Critic networks, which I called dual heuristic programming (DHP) and globalized DHP [15]–[18]. Finally, in 1987, in [19], I proposed a more comprehensive ADP-based architecture for intelligent control and for understanding of the intelligence of the brain. That paper led to a meeting between Sutton and myself that year, bringing together the ADP and the RL schools of research for a time, to some degree.

The Critic in the DHP system outputs a vector of value signals, which are essentially the same as what economists call "shadow prices," except that they are valid in the stochastic case; this turns out to be very convenient in building an interface between automated control systems and human market actors, in the design of the "intelligent electric power grid [14]."

By now, the term "ADP" has become widely disseminated across many disciplines, as we had hoped in organizing the NSF workshops on ADP. Yet, we still need much more mutual cooperation and understanding across disciplines, in order to rise up to the full opportunities illustrated in Fig. 2. We need to overcome the myth that ADP only includes model-free methods, that it cannot handle complex problems, or that we are stuck forever with the limits of lookup tables or linear methods.

We can use ADP to cope with the numerical challenges of robust nonlinear control (which require solutions of the Bellman equation), but we can also use it to make progress on a new concept of *resilient* control, which is closer to what we see in biology. Biological organisms, such as electric power grids and aircraft in war, must somehow cope with environments that are so challenging that it is impossible to guarantee survival or stability, under a truly realistic model of the hazards out there in the environment. The challenge is how to *maximize the long-term probability of survival*, which is a stochastic optimization problem that is well suited for ADP thinking and requires an adaptive approach. ADP thinking turns out to be crucial even in realms such as energy policy and space policy, but that is beyond the scope of this special issue.

PAUL J. WERBOS
National Science Foundation
Arlington, VA 22230 USA

REFERENCES

[1] M. Minsky, *Computers and Thought*, E. A. Feigenbaum and J. Feldman, Eds. New York: McGraw-Hill, 1963.

[2] A. H. Klopf, *The Hedonistic Neuron: A Theory of Memory, Learning and Intelligence*. Washington, DC: Hemisphere, 1982.

[3] J. Von Neumann and O. Morgenstern, *The Theory of Games and Economic Behavior*. Princeton, NJ: Princeton Univ. Press, 1953.

[4] R. Howard, *Dynamic Programming and Markov Processes*. Cambridge, MA: MIT Press, 1960.

[5] P. Werbos, *The Elements of Intelligence*. Namur, Belgium: Cybernetica, 1968. No. 3. [Online]. Available: www.werbos.com

[6] P. Werbos, "Using adaptive dynamic programming to understand and replicate brain intelligence: The next level design," in *Neurodynamics of Higher-Level Cognition and Consciousness*, R. Kozma, Ed. Berlin, Germany: Springer-Verlag, 2007. [Online]. Available: http://arxiv.org/abs/q-bio/0612045

[7] P. Werbos, "Backwards differentiation in AD and neural nets: Past links and new opportunities," in *Automatic Differentiation: Applications, Theory and Implementations*, H. Martin Bucker, G. Corliss, P. Hovland, U. Naumann, and B. Norris, Eds. New York: Springer-Verlag, 2005. [Online]. Available: www.werbos.com

[8] P. J. Werbos, "Beyond regression: New tools for prediction and analysis in the behavioral sciences," Ph.D. dissertation, Committee Appl. Math., Harvard Univ., Cambridge, MA, 1974.

[9] P. J. Werbos, *The Roots of Backpropagation: From Ordered Derivatives to Neural Networks and Political Forecasting*. Hoboken, NJ: Wiley, 1994.

[10] A. R. Barron, "Universal approximation bounds for superpositions of a sigmoidal function," *IEEE Trans. Inf. Theory*, vol. 39, no. 3, pp. 930–945, May 1993.

[11] K. Narendra and S. Mukhopadhyay, "Intelligent control using neural networks," in *Intelligent Control Systems*, M. Gupta and N. Sinha, Eds. Piscataway, NJ: IEEE Press, 1996.

[12] R. S. Sutton, "Learning to predict by the methods of temporal differences," *Mach. Learn.*, vol. 3, no. 1, pp. 9–44, Aug. 1988.

[13] D. A. White and D. A. Sofge, Eds., *Handbook of Intelligent Control*. New York: Van Nostrand, 1992.

[14] *Handbook of Learning and Approximate Dynamic Programming*, J. Si, A. G. Barto, W. B. Powell, and D. Wunsch, Eds. Piscataway, NJ: IEEE Press, 2004.

[15] P. J. Werbos, "Changes in global policy analysis procedures suggested by new methods of optimization," *Policy Anal. Inf. Syst.*, vol. 3, no. 1, pp. 27–52, Jun. 1979.

[16] P. J. Werbos, "Applications of advances in nonlinear sensitivity analysis," in *Proc. IFIP Conf.* , 1981, pp. 762–770.

[17] P. J. Werbos, "Applications of advances in nonlinear sensitivity analysis," in *System Modeling and Optimization*, R. Drenick and F. Kozin, Eds. Berlin, Germany: Springer-Verlag, 1982.

[18] "Advanced forecasting for global crisis warning and models of intelligence," in *General Systems Yearbook*, 1977.

[19] P. J. Werbos, "Building and understanding adaptive systems: A statistical/numerical approach to factory automation and brain research," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-17, no. 1, pp. 7–20, Jan./Feb. 1987.

Paul J. Werbos (M'95–SM'96–F'05) received two degrees in economics from Harvard University, Cambridge, MA, and the London School of Economics, London, U.K.

He is currently the Program Director for computational intelligence with the National Science Foundation, Arlington, VA, and seeks more proposals in that area. In his 1967 paper in *Cybernetica*, he first proposed the idea of approximating dynamic programming as a way to improve reinforcement learning, which is the key theme of the new book *Handbook of Learning and Approximate Dynamic Programming* (Wiley-IEEE Press, 2004).

Dr. Werbos represents the Computational Intelligence Society on the IEEE-USA Energy Policy Committee and serves on the governing boards of International Neural Network Society, the IEEE Industrial Electronics Society, and the Millennium Project of the United Nations University, Tokyo, Japan. He was the recipient of the IEEE Neural Net Pioneer Award for the original invention of backpropagation, in his 1974 Harvard Ph.D. thesis, which was reprinted in his book *The Roots of Backpropagation: From Ordered Derivatives to Neural Networks and Political Forecasting* (Wiley, 1994).