South Asian Journal of Research in Engineering Science and Technology
Volume 3, Issue **10**, **October 2018**

IXplore SAJREST ©2019 ISSN 2455-9261
pp. **36—41**

A Framework for Markov Decision Process

K. Thangamalar,
Lecturer (Sr. Gr.), Mathematics,
Department of Science and Humanities
A.D.J. Dharmambal Polytechnic College
Nagapattinam.

Received June 2018; revised September 2018

ABSTRACT. *In this paper we introduce a modeling paradigm for developing decision process representation for shortest-path problems. Whereas, in previous work attention was restricted to tracking the net using Bellman's equation as a utility function, this work uses a Lyapunov-like function. In this sense, we are changing the traditional cost function by a trajectory-tracking function which is also an optimal cost-to-target function for tracking the net. The main point of the Markov decision process is its ability to represent the system-dynamic and trajectory-dynamic properties of a decision process. Within the system-dynamic properties framework we prove new notions of equilibrium and stability. In the trajectory-dynamic properties framework, we optimize the value of the trajectory-function used for path planning via a Lyapunov-like function, obtaining as a result new characterization for final decision points (optimum points) and stability. Moreover, we show that the system-dynamic and Lyapunov trajectory-dynamic properties of equilibrium, stability and final decision points (optimum points) meet under certain restrictions.*

Keywords: Lyapunov theory, Bellman's equation, Forward Decision Process, Markov decision process.

1. **Introduction.** Hereas previous efforts have restricted attention to track the net using Bellman's equation as a utility function, this paper introduces a modeling paradigm for developing decision process representation, including Markov decision processes (MDP), using a trajectory function as a tool for path planning ([1],[2]).The main point of this paper is its ability to represent the system-dynamic and the trajectory-dynamic properties of a decision process application. We will identify the system dynamic properties as those characteristics related only with the global system behavior, and we will identify the trajectory dynamic properties as those characteristics related with the trajectory function at each state that depends on a probabilistic routing policy.

Within the system-dynamic properties framework we show notions of stability. In this sense, we call equilibrium point to the state in a MDP that does not change, and it is the last state in the net.

In the trajectory-dynamic properties framework we dene the trajectory function as a Lyapunov-like function. By an appropriate selection of the Lyapunov-like function, under certain desired criteria, it is possible to optimize the trajectory. By optimizing the trajectory, we understand that it is maximum or minimum reward (in a certain sense). In addition, we use the notions of stability in the sense of Lyapunov to characterize the stability properties of the MDP. The core idea of our approach uses a non-negative trajectory function that converges in decreasing form to a (set of) final decision states. It is important to point out that the value of the trajectory function associated with the MDP implicitly determines a set of policies, not just a single policy (in case of having several decisions states that could be reached). We call "optimum point" the best choice selected from a number of possible final decision states that may be reached (to select the optimum point, the decision process chooses the strategy that optimizes the reward).

As a result, we extend the system-dynamic framework including the trajectory-dynamic properties. We show that the system-dynamic and the trajectory-dynamic properties of equilibrium, stability and optimum-point conditions converge under certain restrictions: if the MDP is finite then we have that a nal decision state is an equilibrium point.

The paper is structured in the following manner. Section 2 presents the formulation of the decision model, and all the structural assumptions are introduced there. Section 3 discusses the main results of the paper, giving a detailed analysis of the equilibrium, stability and optimum-point conditions for the MDP. Finally, in section 4 some concluding remarks and future work projects are outlined.

2. **Formulation.** The aim of this section is to introduce the decision model and all the structural assumptions related with the Markov model ([3], [5], [9]).

Notation 1: As usual let R be the set of real number and let N be the set of non-negative integers.

Definition 1: A Markov Decision Process is a 5-tuple.

$$MDP = \{S, A, \Upsilon, Q, U\}$$

Where,

S is a countable set of feasible states, $S \epsilon N$, endowed with discrete topology1.

A is the set of actions, which is a metric space. For each $s \epsilon S$; $A(s) \epsilon A$ is the non-empty set of admissible actions at state $s \epsilon S$. Without loss of generality we may take $A=U_{s \epsilon S} A(s)$.

$\upsilon = \{(s;a)|s \epsilon S, a \epsilon A(s)\}$ is the set of admissible state-action pairs, which is a measurable subset of $S \times A$.

$Q = [q_{ij|k}]$ is an array of probabilities, where $q_{ij|k} = P(s_j|s_i,a_k)$ representing the probability associated with the transition from state s_i to state s_j under an action $ak \epsilon A(s_i)$: Note that for any fixed k, $Q|_k$ is a stochastic matrix.

$U : S \rightarrow R_+$ is a trajectory function, associating to each state a real value. Note that U is a function bounded from below. (moreover, it is convenient to use $||U|| = \sup_{s \epsilon S} U(s)$).

3. **Interpretation.** The control model (1) represents a discrete time controlled stochastic system that is observed at time n ϵ N: Denoting by s_n and a_k the state of the system and action applied at time n, respectively, the interpretation of the MDP dynamics is as follows. At each discrete time n ϵ N the state of the system s_n = s ϵ S is observed. For every action an = a ϵ A(s); the probability of the system to find itself in the next state s_{n+1} at time n + 1 is $P(s_{n+1}|sn = s,ak = a)$: Considering the previous states of the trajectory (path, orbit) $(s_0;s_1…,s_n)$ the value of the trajectory function U is obtained and, then the next state s_{n+1} is selected according to U applying some 'criteria'. This is the Markov property of the decision process in (1).

For each n ϵ N the cross product Hn = n S is the set of admissible histories up to time n. The vector hn = (s0;a0;:::;sn1;an1;sn) 2 Hn denotes the history of the process at time n. Considering the previous states of the trajectory (s0;s1;:::;sn), and for every action an 2 A(si); the probability of the system to nd itself in state sj 2 S is qijjk: A policy is a (possibly randomized) measurable rule for choosing actions, which depends on the current state. The policy kji P(ak|si) represents the probability measure associated with the occurrence of an action an from state si. The set of all policies is denoted by Π.

We define a process over S as an finite or infinite sequence of elements of S. If p = (s0;s1:::;sn) is a nite process, we say that sn is the end state of p, and we denote it last(p) = sn. For completeness, first(p) = s0 denote the state in which p starts. Let us dene the sample space = (S A)1; i.e. represents the set of innite processes over S: Let us dene the random variables Xn : ! S for each n 2N, so that we have: Xn(w) = xn for w = (x0;a0;x1;:::).

Let (U, F) be a measurable space with F a -algebra of subsets of the previously defined sample space. We define a probabilistic process over S as a pair (S;P), where P is a probability measure on F. If there is an element s0 2 S such that X0 = s0, we say that s0 is the initial state of the probabilistic process (S,P). Let p = (s0;:::;sn) be a finite process.

We define the likelihood of p by P(p). Intuitively, P(p) is the probability measure of p to occur in an execution of the system. Be aware however that the likelihood function does not dene a probability measure on the set of finite processes, since it does not sum to 1.

Let (S;P) be a probabilistic process, and let p = (s0;:::;s0) be a nite process over S with P(p) > 0. Let us consider the mapping g : p Ʊ.

The mapping g let us dene a probability measure P on (Ʊ;F) as follows: 8A 2 F; P(A) = P(g1(A)jp); where P(jp) is the probability conditional on p. We call the new probabilistic process (S;P) the probabilistic future of process p. We denote by the symbol E the expectation under probability P. By construction, sn = last(p) is the initial state of the probabilistic future of p.

Definition 2: Two given processes p and p0 represent a Path of the following type:

1) OR if one has associated a better probability P to occur at the same time,
2) AND if they have associated any probability P they occur at the same time,
3) Concur if they have associated the same probability P to occur at the same time.

From the previous definition we have the following remark. Remark 1: In a Concur-Path, we have last(p) = last(p0) and therefore we also have P(p) = P(p0).

Consider an arbitrary sj 2 S and for each xed action ak 2A we look at the previous states si of the state sj, denoted by sjk = fsh : h 2 jkg where jk = fh : (sh;ak;sj)g, that materialize the concurrent state-action pair (sh;ak) 2 and form the sum

$$\sum_{h \in \eta_{jk}} \pi_{k|h} q_{hj|k} \; U_h^{(\pi_{k|h})}$$

Notation 2:

$$\left[\begin{array}{c} \sum_{h \in \eta_{jk_0}} \pi_{k_0|h} q_{hj|k_0} U_h, \sum_{h \in \eta_{jk_1}} \pi_{k_1|h} q_{hj|k_1} U_h, \\ \cdots, \sum_{h \in \eta_{jk_f}} \pi_{k_f|h} q_{hj|k_f} U_h \end{array} \right]$$

the index sequence k is the set = fk : ak 2 (sh;ak;sj); and sh running over the set sjkg; and f = #() is the number of actions to state sj: Intuitively, the vector (3) represents all the possible trajectories through the actions ak where (k0;k1;:::;kf) to a state sj for a xed j. Continuing the construction of the definition of the trajectory function U, let us introduce the following denition.

Definition 3: Let MDP = fS;A;;Q;Ug be a Markov Decision Process, let (s0;s1;:::;sn) be a realized trajectory of the system and let L : Rn !R+ be a continuous map. Then L is a Lyapunov-like function [6].

From the previous definition we have the following remark. Remark 2: In the previous definition point 3 we state that L(s) !1when s !1meaningthatthereisno s reachable from some s. Then, formally we define the trajectory function U as follows:

Definition 4: For the discrete time n 2 N the trajectory function U with respect a Markov Decision Process MDP = fS;A;;Q;Ug is represented by

$$U_j = \begin{cases} U_0 & if \;\; n = 0 \\ L(\alpha) & if \;\; n > 0 \end{cases}$$

where

$$\alpha = \left[\begin{array}{c} \sum_{h \in \eta_{jk_0}} \pi_{k_0|h} q_{hj|k_0} U_h, \sum_{h \in \eta_{jk_1}} \pi_{k_1|h} q_{hj|k_1} U_h, \\ \cdots, \sum_{h \in \eta_{jk_f}} \pi_{k_f|h} q_{hj|k_f} U_h \end{array} \right]$$

the function L : D Rn + ! R+ is a function that optimizes the reward through all possible transitions (i.e. trough all the possible trajectories dened by the different ak's), D is the decision set formed by the ks : 0 kl f of all those possible transitions (sh;ak;sj), jk is the index sequence of the list of previous places to sj through action ak and sh (h 2 jkl) is a

specic previous place of sj through action ak.

Explanation. Intuitively, a Lyapunov-like function can be considered as routing function and optimal cost function. In our case, an optimal discrete problem, the cost-to-target values are calculated using a discrete Lyapunov-like function. Every time a discrete vector eld of possible actions is calculated over the decision process. Each applied optimal action (selected via some 'criteria') decreases the optimal value, ensuring that the optimal course of action is followed and establishing a preference relation. In this sense, the criteria change the asymptotic behavior of the Lyapunov-like function by an optimal trajectory tracking value. It is important to note, that the process finishes when the equilibrium point is reached. This point determines a significant difference to the use of Bellman's equation.

Theorem 1. Let MDP = {S;A;;Q;U} be a Markov Decision Process. If s is an equilibrium point then it is a final decision point.

Proof. Let us suppose that s is an equilibrium point we want to show that its trajectory function value has asymptotically approached an infimum (or reached a minimum). Since s is an equilibrium point, by definition, it is the last state of the net. But, this implies that the routing policy attached to the transition(s) that follows s is 0, (in case there is such a transition(s) i.e., worst case). Therefore, its value can not be modified and since the trajectory function is a decreasing function of si an infimum or a minimum is attained. Then, s is a final decision point.

Theorem 2. Let MDP = {S;A;;Q;U} be a (finite) Markov Decision Process (unless s is an equilibrium point). If sf is a final decision point then it is an equilibrium point.

Proof. If sf is a final decision point, since the MDP is finite, there exists some n such that U(sf) = C. Let us suppose that sn is not an equilibrium point.

Corollary 1: Let MDP = {S;A;;Q;U} be a finite Markov Decision Process (unless s is an equilibrium point). Then, an optimum point s4 is an equilibrium point.

Proof: From the previous theorem we know that a final decision point is an equilibrium point and since in particular s4 is final decision point then, it is an equilibrium point.

The finite condition over the MDP can not be relaxed. Let us suppose that the MDP is not finite, i.e. s is in a cycle then, the Lyapunov-like function converges when n $\rightarrow$ 1, to zero i.e., L(s) = 0 but the MDP has no final state therefore, it is not an equilibrium point.

3. **Conclusions.** A formal framework for decision process has been presented. Stability theory was used to characterize the dynamical behavior of the MDP. In addition, we show that the MDP mark-dynamic and trajectory-dynamic properties of equilibrium, stability and

optimum point converge under some mild restrictions. There are a number of questions relating classical planning, that may in the future be addressed satisfactorily within this approach.

REFERENCES

[1] J. Clempner. Towards modeling the shortest-path problem and games with petri nets. Proc. Of the doctoral consortium at the icatpn, 1-12, 2006.

[2] J. Clempner and j. Medel. A simple modal approach to decision process. Proceedings of the 9th wseas int.conf. On mathematical and computational methods in science and engineering, 34-38, 2007.

[3] O. Hernændez-lerma and j.b. Lasserre. Discrete-time markov control process: basic optimality criteria. Berlin, germany : springer, 1996.

[4] O. Hernændez-lerma, g. Carrasco and r. Pøre-hernændez. Markov control processes with the expected total cost criterion: optimality, stability and transient model. Acta applicadae matematicae, 59, 3 229269, 1999.

[5] O. Hernændez-lerma and j.b. Lasserre. Futher topics on discrete-time markov control process. Berlin, germany: springer-verlag, 1999.

[6] R. E. Kalman and j. E. Bertram. Control system analysis and design via the "second method" of lyapunov. Journal of basic engineering, 82(d), 371-393, 1960.

[7] Lakshmikantham, s. Leela and a.a. Martynyuk, practical stability of nonlinear systems, world scientic, singapore, 1990.

[8] V. Lakshmikantham, v.m. Matrosov and s. Sivasundaram, vector lyapunov functions and stability analysis of nonlinear systems, kluwer academic publ., dordrecht, 1991.

[9] A. S. Poznyak, k. Najim and e. Gomez-ramirez. Self-learning control of nite markov chains. Marcel dekker, inc., new york, 2000.