Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 50-1 (2017) 8435–8440

# A Consensus Approach to Dynamic Programming

**Mattia Laurini** * **Luca Consolini** * **Marco Locatelli** *

* *Dipartimento di Ingegneria e Architettura, Università degli Studi di Parma, Parco Area delle Scienze 181/a, 43124 Parma, Italy (e-mail: mattia.laurini@studenti.unipr.it, {luca.consolini, marco.locatelli}@unipr.it).*

**Abstract:** Motivated by the finite element formulation of the Hamilton-Jacobi-Bellman (HJB) equation, we introduce a consensus algorithm to compute the solution of a class of optimization problems that can be solved with a fixed point iteration. The proposed algorithm reduces the computational cost in terms of elementary operations with respect to a complete fixed point iteration. We provide theoretical results on maximum error rate and on the convergence of the algorithm. As an application, we compute the minimum-time solution for a parking maneuver of a car-like vehicle, comparing the fixed point iteration with the consensus iteration.



*Keywords:* dynamic programming, communication networks, directed graphs, optimization problems, nonlinear systems, numerical methods, convergence proofs.

## 1. INTRODUCTION

Consider a control system of the form

$$\begin{cases} \dot{x}(t) = f(x(t), u(t)) \\ x(0) = x_0, \end{cases}$$

where $f$ is a continuous function, $x_0$ is the initial state, $u(t) \in U$ is the control input and $U$ is a compact set of admissible controls. A fundamental problem in control consists in finding the control function $u$ that minimizes the infinite horizon cost functional

$$J_{x_0}(u) := \int_0^\infty l(x(t), u(t))dt.$$

where $l$ is a continuous cost function.

A general method for addressing this problem is given by the Hamilton-Jacobi-Bellman (HJB) equation, which is considered by many works in control applications, such as Bertsekas (1995), Grüne and Semmler (2004) or Luus (1994). In general, it is not possible to obtain a closed form solution of the HJB equation because it is a nonlinear partial differential equation. Thus, many works provide numerical procedures, see for instance Wang et al. (2000), Liu and Wei (2013), Al-Tamimi et al. (2008), Bian et al. (2014) or Jiang and Jiang (2014). As can be seen in Appendix A of Bardi and Capuzzo-Dolcetta (2008), we will briefly recall that, under some approximations, it is possible to obtain the solution of the HJB equation as the limit for $k \to \infty$ of a fixed point iteration of the form

$$\begin{cases} x(k+1) = \min_{p=1,\dots,M} \{A_p x(k) + b_p\} \\ x(0) = x_0, \end{cases} \tag{1}$$

where, for $p = 1, \dots, M$, $A_p$ are sparse positive matrices and $b_p$ are positive vectors. Note that, in (1), the minimum operation is performed component-wise, (i.e., for each component of the vector $A_p x(k) + b_p$, a different value of $p$ can be chosen).

It is well known that the convergence rate of iteration (1) is rather poor. Thus, in literature, some works provide an acceleration policy, that is, a second iteration that converges faster to the same limit as (1). As an example, Anderson (1965) presents such an iteration for general fixed point iterations; whilst Alla et al. (2015) and Laurini et al. (2016) introduce acceleration policies designed specifically for (1).

In this paper, we present an algorithm inspired by broadcast-based consensus aimed at reducing the computational cost of iteration (1) in terms of row-column product operations. In fact, when applying iteration (1), all coordinates values of $x$ are updated even though it many not be necessary. Considering the coordinates of $x$ as the nodes of a communication graph, the proposed algorithm updates only those nodes whose neighbors have undergone a sufficiently large variation in their value in the previous iteration. Namely, the nodes communicate their value to their neighbors only when the variation they have undergone is relevant. This is somehow similar to broadcast-based consensus algorithms, such as those presented in Aysal et al. (2009) and Franceschelli et al. (2010).

The work is organized as follows: in section 2 we introduce the problem of the numerical solution of HJB equation and show that it is an instance of problem (1), in section 3 we formulate the consensus iteration, in section 4 we state the main result, in section 5 we present both the algorithm based on fixed point iteration and on consensus iteration, and in Appendix A we provide the proof of convergence of the introduced algorithm. As an example, in section 6, we show the computational advantages of the algorithm in computing the minimum-time solution of a parking maneuver for a car-like vehicle with bounded velocity and steering angle.

**Notation.** By $\mathbb{R}_+$ we denote the interval $[0, +\infty)$. Let $x \in \mathbb{R}^N$ and $A \in \mathbb{R}^{N\times M}$, $\forall\, i \in \{1, \dots, N\}$ we denote the $i$-th component of $x$ with $[x]_i$ and the $i$-th row of $A$ with $[A]_{i:}$. Further, $\forall\, j \in \{1, \dots, M\}$ we denote the $j$-th column of $A$ with $[A]_{:j}$. Function $\|\cdot\| : \mathbb{R}^N \to \mathbb{R}_+$ is the infinity norm, namely the maximum norm, of $\mathbb{R}^N$ (i.e., $\forall\, x \in \mathbb{R}^N$ $\|x\| = \max_{i=1,\dots,N} |[x]_i|$); $\|\cdot\|$ is also used to denote the induced matrix norm. Symbol $\varnothing$ denotes the empty set. A directed multigraph $\mathscr{G}$ is a triple $(\mathscr{V}, \mathscr{E}, f)$, where $\mathscr{V}$ is a set whose elements are called nodes (or vertices), $\mathscr{E}$ a set of ordered pairs of nodes of $\mathscr{V}$ called edges and $f : \mathscr{E} \to \mathscr{V}^2$ is


10.1016/j.ifacol.2017.08.735

a mapping assigning to every edge its ends. Note that in direct multigraphs it may be that

$$\exists\, e_1, e_2 \in \mathscr{E} : (e_1 \neq e_2 \,\wedge\, f(e_1) = f(e_2))$$

or

$$\exists\, e \in \mathscr{E} : \exists\, i \in \mathscr{V} : f(e) = (i,i),$$

that is, multiple edges (edges sharing the same ends) are allowed as well as self-loops (edges whose ends coincide with a single vertex). As an example one can refer to Figure 1 which represents a four nodes direct multigraph $\mathscr{G} = (\mathscr{V}, \mathscr{E}, f)$, where $\mathscr{V} = \{1,2,3,4\}$, with multiple edges and self-loops.

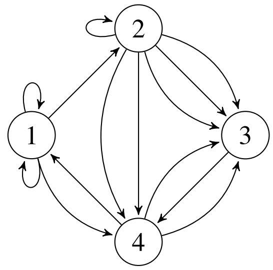


Fig. 1. Directed multigraph with four nodes.

Given a directed multigraph $\mathscr{G} = (\mathscr{V}, \mathscr{E}, f)$, let us define, $\forall\, i \in \mathscr{V}$, the set of direct successors of node $i$

$$\mathscr{N}(i) := \{ j \in \mathscr{V} : \exists\, e \in \mathscr{E} : f(e) = (i,j)\}$$

and the set of direct predecessors of node $i$

$$\mathscr{N}^{-1}(i) := \{ j \in \mathscr{V} : \exists\, e \in \mathscr{E} : f(e) = (j,i)\}.$$

## 2. PROBLEM MOTIVATION

In this section we show that problem (1) is associated to the numerical solution of the HJB equation. For an extensive discussion, see Appendix A of Bardi and Capuzzo-Dolcetta (2008). Let us consider a control system defined by the following differential equation in $\mathbb{R}^n$:

$$\begin{cases} \dot{x}(t) = f(x(t), u(t)) \\ x(0) = x_0, \end{cases}$$

where $f : \mathbb{R}^n \times U \to \mathbb{R}^n$ is a continuous function, $x_0 \in \mathbb{R}^n$ is the initial state, $u = u(t) \in U \subset \mathbb{R}^m$ is the control input and $U$ is a compact set of admissible controls. Let us consider an infinite horizon problem defined by the functional

$$J_{x_0}(u) = \int_0^\infty l(x(t), u(t)) e^{-\lambda t} dt, \tag{2}$$

where $l : \mathbb{R}^n \times U \to \mathbb{R}$ is a continuous cost function and constant $\lambda > 0$ is the viscosity parameter. Let us define the value function $v : \mathbb{R}^n \to \mathbb{R}$ as follows:

$$v(x_0) = \inf_{u \in U} J_{x_0}(u).$$

We refer to Bardi and Capuzzo-Dolcetta (2008) for a proof of the fact that the value function $v$ is the unique viscosity solution of HJB equation:

$$\lambda v(x) + \sup_{u \in U}\{-Dv(x) f(x,u) - l(x,u)\} = 0, \quad x \in \mathbb{R}^n, \tag{3}$$

where $Dv$ denotes the gradient of $v$.

Equation (3), in general, does not admit a closed form solution, so one needs to apply a numerical method in order to compute an approximate solution. For instance, the scheme presented in Bardi and Capuzzo-Dolcetta (2008) relies on a finite dimensional approximation of state and control spaces and a discretization in time. More precisely, in (3) one can approximate $Dv(x)f(x,u) \simeq h^{-1}(v(x + hf(x,u)) - v(x))$, where $h > 0$ represents an integration time. So by approximating $(1+\lambda h)^{-1} \simeq (1 - \lambda h)$, $(1+\lambda h)^{-1} h \simeq h$ one obtains the HJB equation in discrete time:

$$v_h(x) = \min_{u \in U}\{(1-\lambda h) v_h(x + hf(x,u)) + hl(x,u))\}, \quad x \in \mathbb{R}^n.$$

Moreover, if one considers a triangulation on a finite set of vertices $S = \{x_i\} \subset \mathbb{R}^n$, $i = 1, \ldots, N$, function $v$ can be approximated by a linear function of the finite set of variables $v_h(x_i)$, $i = 1, \ldots, N$. It is also possible to discretize the control space, substituting $U$ with a finite set of controls $\{u_1, \ldots, u_M\}$, obtaining eventually the following formulation

$$\begin{aligned} v_h(x_i) = \min_{u_p}\{(1-\lambda h) v_h(x_i + hf(x_i,u_p)) + hl(x_i,u_p))\}, \\ i = 1, \ldots, N \text{ and } p = 1, \ldots, M. \end{aligned} \tag{4}$$

For a wider and more detailed dissertation see Bardi and Capuzzo-Dolcetta (2008).

Set vector $x^\star := [v_h(x_1), v_h(x_2), \ldots, v_h(x_N)]$, in this way $x^\star \in \mathbb{R}^N$ represents the value of the cost function on the grid points. Note that, for each $x_i, u_p$, the right-hand side of (4) is affine with respect to $x^\star$, so that problem (4) can be rewritten in form

$$x^\star = \min_{p=1,\ldots,M}\{A_p x^\star + b_p\}, \tag{5}$$

where, for $p = 1, \ldots, M$, $A_p \in \mathbb{R}_+^{N \times N}$ are suitable nonnegative matrices and $b_p \in \mathbb{R}_+^N$ are suitable nonnegative vectors. Define map $T : \mathbb{R}_+^N \to \mathbb{R}_+^N$ as

$$T(x) := \min_{p=1,\ldots,M}\{A_p x + b_p\}. \tag{6}$$

In Bardi and Capuzzo-Dolcetta (2008) it is shown that $T$ is a contraction, so (5) can be solved as a fixed point iteration. Namely, setting

$$\begin{cases} x(k+1) = T(x(k)), \\ x(0) = x_0, \end{cases} \tag{7}$$

the solution $x^\star$ of (5) is obtained as

$$x^\star = \lim_{k \to \infty} x(k),$$

for any initial condition $x_0$.

## 3. PROBLEM FORMULATION

Our aim is to provide a procedure that allows reducing the computational cost of iteration (7) in terms of row-column product operations. When applying iteration (7), the value of all nodes is updated even though many nodes values may undergo a negligible variation. So the main idea is to update only those nodes whose at least one of the direct predecessors has undergone a sufficiently large variation in its value in the previous iteration. To this end, we may consider the vertices indices of triangulation $S$ as the nodes of a multigraph. Let $\mathscr{G} = (\mathscr{V}, \mathscr{E}, f)$ be a direct multigraph with $\mathscr{V} := \{1, \ldots, N\}$ and $\mathscr{E}$ satisfying the following statement: $\forall\, p = 1, \ldots, M, \forall\, i, j \in \mathscr{V}$

$$[A_p]_{ij} \neq 0 \Leftrightarrow \exists\, e \in \mathscr{E} : f(e) = (i,j).$$

Given a tolerance $\varepsilon > 0$, let us define $H_\varepsilon : \mathbb{R}_+^N \times \mathbb{R}_+^N \to \mathbb{R}_+^N$ as follows: $\forall\, x, y \in \mathbb{R}_+^N, \forall\, i \in \mathscr{V}$

$$[H_\varepsilon(x,y)]_i := \begin{cases} [T(y)]_i, & \text{if } \exists j \in \mathscr{N}^{-1}(i) : \left|[y]_j - [x]_j\right| > \varepsilon \\ [y]_i, & \text{otherwise,} \end{cases} \tag{8}$$

where $T$ is defined in (6). And consider the following sequence