21st International Symposium on
Mathematical Theory of Networks and Systems
July 7-11, 2014. Groningen, The Netherlands

Domain Decomposition based Parallel Howard's Algorithm

Adriano Festa

The Howard's algorithm (also called policy iteration algorithm) is a classical method for the resolution of a discrete Hamilton-Jacobi equation. This technique, developed by Bellman and Howard [3], [8], is of large use in applications because of the high efficiency and performances. The main feature of this method is that, in presence of a finite number of controls, the algorithm reaches the solution in finite number of iterations [4]; this gives to the method a special interest.

The problem considered is the following. Let Ω be an open domain of $\mathbb{R}^d$ ($d \geq 1$); the steady *Hamilton-Jacobi equation* (HJ) is described in the following form:

$$\begin{cases} \lambda v(x) + H(x, Dv(x)) = 0 & x \in \Omega \\ v(x) = g(x) & x \in \partial\Omega \end{cases} \tag{1}$$

where $\lambda \in \mathbb{R}^+$ is the discount factor, $g : \Omega \to \mathbb{R}$ is the exit cost, and the Hamiltonian $H : \Omega \times \mathbb{R}^d \to \mathbb{R}$ is defined by: $H(x,p) := \inf_{\alpha\in\mathcal{A}}\{-f(x,\alpha)\cdot p - l(x,\alpha)\}$ with $f : \Omega \times \mathcal{A} \to \mathbb{R}$ (dynamics) and $l : \Omega \times \mathcal{A} \to \mathbb{R}$ (running cost). Under classical assumptions on the data, it is known (see e.g. [1], II.3) that the equation (1) admits a unique solution $v : \overline{\Omega} \to \mathbb{R}$, and v is the value function to the infinite horizon problem with exit cost, where τ_x is the *first time of exit* form Ω:

$$v(x) = \inf_{a(\cdot)} \int_0^{\tau_x(a)} l(y_x(s), a(s)) e^{-\lambda s} ds + e^{-\lambda\tau_x(a)} g(y_x(\tau_x(a)))$$

Where $a(\cdot) \in L^\infty([0, +\infty[; \mathcal{A})$ and $y_x(t)$ is a solution of

$$\begin{cases} \dot{y}(t) = f(y_x(t), a(t)), \\ y_x(0) = x. \end{cases}$$

The numerical discretisation of equation (1) by a monotone and consistent scheme (like Semi-Lagrangian [6]V.2, or Finite Differences[5]) leads to a nonlinear system in the following form:

$$\text{Find } V \in \mathbb{R}^N; \quad \min_{\alpha\in\mathcal{A}^N}(B(\alpha)V - c(\alpha)) = 0 \tag{2}$$

where α is the control, and B is a $N \times N$ matrix and c is a N vector, the unknown vector V is an approximation of the value function v on a given grid.

The *Policy Iteration Algorithm* (or Howard's Algorithm) consists in a two-steps iteration with an alternating improvement of the policy and the value function:

Let $\alpha^0 \in \mathcal{A}^N$ be an initial policy. For for $n \geq 0$ compute:

- $X^{n+1} \in \mathbb{R}^N$ s.t. $B(\alpha^n)X^{n+1} - c(\alpha^n) = 0$
(Value evaluation step)
- $\alpha^{n+1} = \arg\min_{\alpha\in\mathcal{A}^N}\left[B(\alpha)X^{n+1} - c(\alpha)\right]$
(Policy evaluation step)

It is by now known that under a monotonicity assumption on the matrices $B(\alpha)$, the above algorithm is a nonsmooth Newton method that converges to the solution of problem (2). Moreover, if $\mathcal{A}$ has a finite number of elements, then the algorithm converges in a finite number of iterations.

I. Domain Decomposition and Parallel version

The use of the Howard's algorithm is strongly limited by restriction on memory storage, especially in the case of high dimensional spaces, which is the typical situation when the nonlinear system (2) comes from the discretization of a d-dimensional HJ equation (1). For this reason, there a wide interest for a parallel version of it.

Our proposal consists in combining the policy iteration algorithm with a domain decomposition principle for HJ equations. Using the new theoretical framework of HJ equations on submanifolds [9], we consider a decomposition of Ω on a collection of subdomains:

$$\Omega := \cup_{i=1}^{M_\Omega}\Omega_i \cup_{j=1}^{M_\Gamma}\Gamma_j, \quad \text{with } \mathring{\Omega}_i \cap \mathring{\Omega}_j = \emptyset.$$

ENSTA ParisTech 828, Boulevard des Marchaux, 91120 Palaiseau, festa@ensta.fr

ISBN: 978-90-367-6321-9

Where the interfaces Γ_j, $j = 1, \cdots, M_\Gamma$ are some strata of dimension lower than d defined as the intersection of two subdomains $\Omega_i \cap \Omega_k$ for $i \neq k$. Following [2], the Hamilton-Jacobi equation (1) is equivalent to the following system:

$$\begin{cases} \lambda v(x) + H(x, Dv(x)) = 0, \quad x \in \Omega_i, \\ i = 1, \cdots, M_\Omega. \end{cases} \tag{3}$$

completed by a system of transmission equations on the interfaces

$$\begin{cases} \lambda v(x) + H_j(x, Dv(x)) = 0 \quad x \in \Gamma_j \\ j = 1, \cdots, M_\Gamma. \end{cases} \tag{4}$$

for adequately well defined Hamiltonians.

The technique that is presented here is related to the numerical resolution of the decomposed system above. The talk will present first some theoretical results to justify the domain decomposition technique. Then, a parallel version of Howard's algorithm based on the domain decomposition will be discussed and analysed in connection to some hybrid systems. This algorithm consists of two steps iterations: (i) Use Howard's algorithm to solve in parallel on n threads the nonlinear systems obtained after discretization of (3) on the subdomains Ω_i (in this step the values of V are fixed on the boundaries); (ii) Update the values of V. on the interfaces by using Howard's algorithm on the nonlinear system obtained after discretization of the HJ equations (4) on Γ_j.

Under some specific assumptions of monotonicity it is possible to prove the convergence of the algorithm to the numerical solution of the original problem (2) [7].

II. Examples and Tests

The performances of the algorithm and its characteristics as speeding up technique can be tested and shown. Here it is considered the case of an eikonal equation in a 1D and 2D space, in both the cases the problem models the distance function from the boundary of an open set, $(-1, 1)$ in the first case, $(-1, 1) \times (-1, 1)$ in the second one. As we can see in Figure 2 and in Table I the improvements in terms of efficiency of the presented technique are significant. In the table we compare the time necessary to reach the approximated solution, analysing the various phases of the algorithm: the time necessary to solve every sub-problem (first column), for the iterative part (which passes the information through the threads, next column), finally

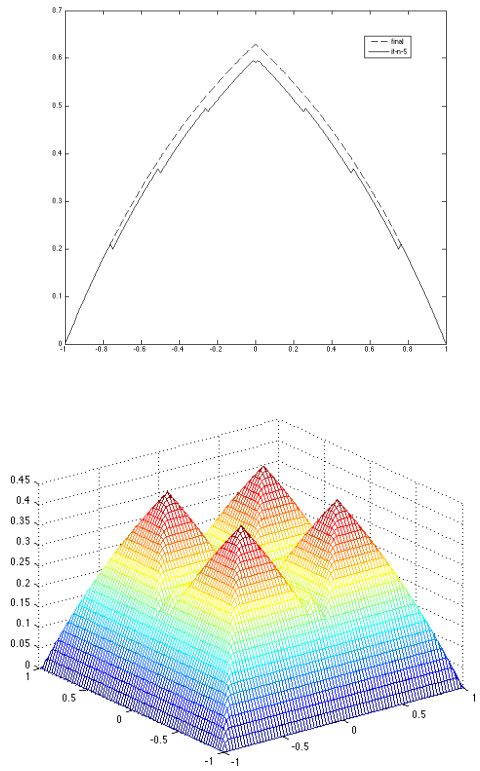

Fig. 1. Approximated solution of the iterative/parallel algorithm (left) in the 1D case, final time (dotted) and fifth iteration (solid), in the 2D case (right, 20th iteration).

the total time. It is highlighted the optimal choice of number of threads (which is different with respect of the problem and discretization step) which has to be found with a tuning procedure. In Table II is presented the same comparison in the case of dimension two, comparing the complexity for various discretization steps.

This technique is shown to be extended also to some others cases of interest: *Target problems*, *Obstacle Problems* and *Min-Max Problems*. An example of Target Problem, well known benchmark in the field, is the so-called *Zermelo's navigation problem*. The main difficulty, in this example, is that the dynamic is driven by a force of comparable power with respect to our control. The target to reach will be a ball of radius equal to 0.005 centred in the origin

$$f(x, a) = a + \begin{pmatrix} 1 - x_2^2 \\ 0 \end{pmatrix}, \quad \Omega = [-1, 1]^2,$$

TABLE I

TESTING PERFORMANCES, 1D. OUR METHOD COMPARED WITH THE CLASSIC HOWARD'S WITH VARIOUS NUMBER OF THREADS

dx=0.0125	Classic HA	Parallel Howard's Algorithm		
threads	time (s)	(par.)	(it.)	Total time
2		0.48	1e-4	0.36
4		8e-3	1e-4	0.086
8	0.32	18e-4	6e-4	0.014
16		7e-4	4e-4	0.0095
32		2e-4	6e-3	0.011

TABLE II

TESTING PERFORMANCES, 2D. COMPARISON WITH CLASSICAL METHOD AND PH WITH 4 THREADS

dx=0.0125	Classic HA	Parallel Howard's Algorithm		
dx	time (s) .	tm (par.)	tm (it.)	Total tm
0.1	0.05	0.009	0.02	0.04
0.05	2.41	0.05	0.03	0.14
0.025	73.3	2.5	0.15	7.83
0.0125	-	76	1.293	383.3

TABLE III

ZERMELO'S NAVIGATION PROBLEM. COMPARISON OF VARIOUS CHOICES OF THE NUMBER OF THREADS

dx=0.0125	Classic HA	Parallel Howard's Algorithm		
threads	time (s)	tm (par.)	tm (it.)	Total time
4		1.31	0.13(4)	5.4
9		0.7	0.7(5)	4.2
16	37.9	0.031	1.38(5)	1.53
25		0.02	2.7(6)	3.9
36		0.01	5.19(7)	5.28

and $A = B(0,1)$, $\lambda = 1$, $l(x,y,a) = 1$.

In Table III a comparison with the number of threads chosen is made. Now we are in presence of characteristics not aligned with the grid, but the performances of the method are poorly effected. Convergence is archived with performances absolutely comparable with the ones already described for the Eikonal Equation.

ACKNOWLEDGEMENTS

I would like to thank Hasnaa Zidani of the UMA Laboratory of ENSTA for the support in developing the subject.
This work was partially supported by the European Union under the 7th Framework Programme FP7-PEOPLE-2010-ITN SADCO, Sensitivity Analysis for Deterministic Controller Design.

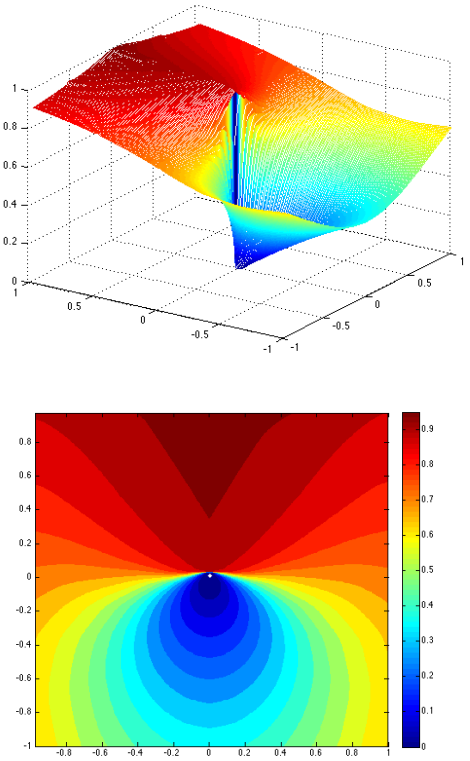

Fig. 2. Approximated solution for the Zermelo's navigation problem $dx = 0,01$.

REFERENCES

[1] M. BARDI AND I. CAPUZZO-DOLCETTA, *Optimal Control and Viscosity Solution of Hamilton-Jacobi-Bellman Equations*. Birkhauser, Boston Heidelberg, 1997.
[2] G. BARLES, A. BRIANI AND E. CHASSEIGNE, A Bellman approach for two-domains optimal control problems in $\mathbb{R}^N$. ESAIM Contr. Op. Ca. Va., Vol. 19 n. 03 (2013), pp. 710–739.
[3] R. BELLMAN, *Dynamic Programming*, Princeton University Press, Princeton, NJ, 1957.
[4] O. BOKANOWSKI, S. MAROSO AND H. ZIDANI, Some convergence results for Howard's algorithm, SIAM J. Numer. Anal. Vol 47, n. 4, (2009), pp. 3001-3026.
[5] M. G. CRANDALL AND P. L. LIONS Two Approximations of Solutions of Hamilton-Jacobi Equations, Math. Comp. Vol. 43 n. 167 (1984) pp. 1–19.
[6] M. FALCONE AND R. FERRETTI, *Semi-Lagrangian Approximation Schemes for Linear and Hamilton-Jacobi Equations*, Applied Mathematics series, SIAM, 2013.
[7] A. FESTA, Domain Decomposition based Parallel Howard's Algorithm, preprint (2014).
[8] R.A. HOWARD, *Dynamic Programming and Markov Processes*, The MIT Press, Cambridge, MA, 1960.
[9] Z. RAO AND H. ZIDANI, Hamilton-Jacobi-Bellman Equations on Multi-Domains, Birkhauser Basel. Volume 164, (2013).