

IRRIGATION CANALS DISTRIBUTED MODEL-BASED PREDICTIVE CONTROL USING MULTI-AGENT SYSTEMS

Clément Raievsky, van Thang Pham, Laurent Lefevre, Jean-Paul Jamont

▶ **To cite this version:**

Clément Raievsky, van Thang Pham, Laurent Lefevre, Jean-Paul Jamont. IRRIGATION CANALS DISTRIBUTED MODEL-BASED PREDICTIVE CONTROL USING MULTI-AGENT SYSTEMS . MOSIM 2014, 10ème Conférence Francophone de Modélisation, Optimisation et Simulation, Nov 2014, Nancy, France. hal-01166605

HAL Id: hal-01166605
https://hal.science/hal-01166605

Submitted on 23 Jun 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

10^{th} *International Conference on MOdeling, Optimization and SIMlation - MOSIM14 - November 5-7-2014- Nancy - France "Toward circular Economy"*

Irrigation Canals Distributed Model-Based Predictive Control Using Multi-Agent Systems

C. Raïevsky, V. T. Pham, J.-P. Jamont, L. Lefèvre

Univ. Grenoble Alpes, LCIS, F-26902 Valence, France
{clement.raievsky,van-thang.pham,jean-paul.jamont,laurent.lefevre}@lcis.grenoble-inp.fr

ABSTRACT: *Our work is motivated by the application of the so-called Model-based Predictive Control (MPC) in the context of irrigation networks. The main obstacles for applying classical centralized methods to such complex systems are the huge computational effort required to solve the corresponding optimization problems and the lack of flexibility in the problem structure's description. In this work we propose the application of the multiagent paradigm to a distributed scheme of MPC. The global optimization problem is firstly decomposed into sub-problems which are then solved in parallel by agents. The communication between agents is then used to guarantee the convergence of this distributed scheme to the global optimum. In addition, agents are given the ability to rearrange and reconfigure themselves in order to improve the fault tolerance of the control scheme. Finally, the proposed approach is validated in a simulated irrigation system made of interconnected canal reaches interconnected through actuated gates.*

KEYWORDS: *Irrigation canals; Decentralized model-based predictive control; Multiagent systems*

1 INTRODUCTION

Water is a critical resource for our survival and irrigation is one of the main stress put on this resource. That is why designing efficient control mechanisms for water systems is crucial. Those systems are complex collections of interconnected water bodies (e.g. lakes, reservoirs), natural canals, and pipes. Gates, dams, pumps, and valves are used to control water flows and achieve proper delivery of irrigation water.

Controlling such heterogeneous and spatially distributed systems is very challenging using centralized techniques due to communication constraints, computational cost and the required adaptivity.

Free surface canals, in which the water flow dynamics are usually modeled using the Saint-Venant (or Shallow Water) partial differential equations (PDEs), are commonly the main components used for long range water transportation in irrigation networks. The control design for such systems is generally based on one of the two following approaches. The first one, called the indirect approach, begins with an approximation of the PDEs by ordinary differential equations (ODEs) to which a finite-dimensional control synthesis is applied. The advantage of this approach is the availability of control synthesis techniques for ODEs.

Finite dimensional approximations for these dynamics often result in meaningless qualitative dynamical properties for the approximated solution. This usually motivates the "direct approach" where the control design is directly derived from the infinite dimensional model. The obtained control law is numerically approximated only at the implementation stage.

Our objective here is to study the direct approach for linearized Saint-Venant equations. This approach can however be applied to more general hyperbolic systems of conservation laws used in several fields such as gas dynamics (Serre 1999), road traffic (Colombo, Goatin & Rosini 2011), air traffic (Bayen, Raffard & Tomlin 2006), transport-reaction processes (Dubljevic, Mhaskar, El-Farra & Christofides 2005), and pressurized water transportation systems (Georges 2009).

We are particularly interested in the model predictive control (MPC - also known as receding horizon control), in which the control action is obtained by solving repeatedly, online, a finite horizon open-loop optimal control problem. Among the advantages of MPC, one can mention the ability to obtain a guaranteed stability, to handle constraints, to incorporate forecast information and to minimize a given criterion. The approach was well studied for finite-dimensional systems, even in the nonlinear case (see e.g. (Findeisen, Imsland, Allgöwer & Foss 2003) and (Rawlings & Mayne 2009)). Some extension to infinite-dimensional systems was also investigated, as in (Ito & Kunisch 2002). But the latter work is concerned only with the case of distributed control. In (Christofides & Daoutidis 1997) and (Dubljevic

et al. 2005), the authors proposed MPC approach for parabolic systems but the control synthesis was based on a finite-dimensional approximation of the PDEs. An infinite-dimensional MPC for boundary control of nonlinear Saint-Venant equations was considered in (Georges 2009), and solved by calculus of variations approach. Our recent work ((Pham, Georges & Besançon 2010, Pham, Georges & Besançon 2012)) established the stability of MPC for a single linear hyperbolic system as well as for a cascaded network of such systems.

The usefulness of the MPC approach for these problems related to *interconnected infinite-dimensional systems* is however limited by the required computational effort when the control action for the whole network is calculated in a centralized manner by a single controller. This obstacle can be tackled by using the so-called distributed MPC configuration or decomposition-coordination approach in which the optimization problem of the entire system is divided into several sub-problems, each of them being allocated to a local controller (sometimes referred to as an "agent"). The global optimal control action is then obtained by exchanging information between these agents. This is currently a living topic in the MPC community and several results have been established (see e.g. (Scattolini 2009), (Stewart, Venkat, Rawlings, Wright & Pannocchia 2010), (Christofides, Scattolini & de la Peña 2013) or (Liu, Chen, de la Peña & Christofides 2010)), mostly in the finite-dimensional case, while very few studies ((Georges 2009)) consider the infinite-dimensional case.

In this paper, we consider an algorithm of distributed model predictive control (DMPC) for a system of cascaded reaches of an irrigation canal through the connecting sliding gates. The DMPC was in fact considered in several works for water transportation systems (see (Carpentier & Cohen 1993), (Fawal, Georges & Bornard 1998), (Zarate-Florez, Molina, Besancon & Faille 2012) and (Igreja, Lemos, Cadete, Rato & Rijo 2012)), but with a finite-dimensional model.

In the irrigation canals control problem, the computational costs of centralized algorithms is mainly a financial problem, not a computation time one. The control sampling time is commonly around 5 minutes or above. The need for decentralization is therefore motivated essentially by robustness of the control architecture with respect to actuators/sensors faults, communication breaks, system openness, or runtime modification of control objectives. To address these needs, another main objective of this paper is to explore the combination of the DMPC approach with the *multi-agent* paradigm. Multi-agent systems (MAS) are able to address problems where the structure of the model may change at runtime. MAS can also manage dynamical interactions between the subsytems (agents) or open systems in which components may be added or removed at runtime. This paper shows how combining DMPC with MAS improves the adaptability and fault tolerance of a water management system by making it able to handle a component failure without requiring to redesign the control architecture.

Specifically regarding DMPC, the novelty of our work in comparison with previous ones dealing with irrigation systems (e.g. (Negenborn, van Overloop, Keviczky & Schutter 2009) or (Igreja et al. 2012)) can be summarized in two facts. The first one is the formulation of the DMPC directly in infinite-dimensional setting. The second one is the implementation of this control scheme in a MAS simulation platform which allowed us to change the decomposition of the control scheme at runtime.

The paper is organized as follows. In the first section the considered system of cascaded network of irrigation canal reaches is presented. The principle of centralized MPC and its application to such system is summarized in section 2. In section 3, a decomposition approach called prediction decomposition proposed by are considered and applied to obtain a decentralized MPC scheme. The convergence of these algorithms to the global optimum is discussed. The next section is dedicated to present the general concepts of multi-agent systems, the ASTRO architecture (Occello, Demazeau & Baeijs 1998) and its application in order to improve the adaptability and fault tolerance of the control scheme. Simulation results are presented in Section 6. Some conclusions and future research directions are finally given.

2 IRRIGATION NETWORKS

In this paper, we present an application involving the linearized model of an irrigation canal consisting of N cascaded pools. Each pool is usually described by a set of two partial differential equations (PDEs) named Saint-Venant equations, which represent the mass and the momentum conservation (see (Graf & Altinakar 2000)):

$$\begin{array}{lcl} B_i\partial_t h_i + \partial_x Q_i & = & 0 \\ \partial_t Q_i + \partial_x \left(\frac{Q_i^2}{B_i h_i} + \frac{1}{2}B_i g h_i^2\right) & = & gB_i h_i(I_i - J(Q_i, h_i), \\ (x,t) \in [0,L] \times [0,\infty), & & i = 1, ..., n, \end{array} \tag{1}$$

where h_i denotes the water depth, Q_i the discharge, g the gravitational acceleration, B_i the canal width, I_i the slope and J the friction term.

The friction is modeled by the classical Manning for-

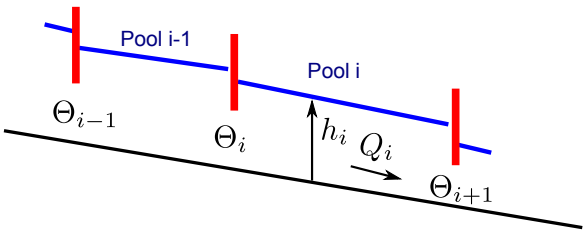


Figure 1 – Multi-pool system

mula ((Graf & Altinakar 2000)):

$$J(h,Q) = \frac{Q^2}{k^2B^2h^2\left(\frac{Bh}{B+2h}\right)^{4/3}}, \tag{2}$$

where k is the Manning-Strickler (MS) coefficient.

Interconnections between pools are subject to a set of $N+1$ sliding (or sluice) gate equations:

$$Q_{g_i} = K_i^2\Theta_i^2(t)2g(h_{us}^{g_i} - h_{ds}^{g_i}), i = 1, ..., N+1, \tag{3}$$

These have been obtained from the linearization of usual submerged gates equations (Graf & Altinakar 2000). Also $N-1$ discharge conservation constraints must hold:

$$Q_i(L,t) = Q_{i+1}(0,t), \quad i = 1, ..., n-1, \tag{4}$$

where Q_{g_i} is the discharge through the gate, K_i the gate coefficient, Θ_i its opening, $h_{us}^{g_i}$ and $h_{ds}^{g_i}$ are the water levels at upstream and at downstream respectively.

Let us now consider the linearization of the system around a uniform steady state $(\bar{h}_i, \bar{Q}_i)$ which has to satisfy $\bar{Q}_i$ = constant and $J(\bar{h}_i, \bar{Q}_i) = I$. Denote by $\tilde{h}_i = h_i - \bar{h}_i$, $\tilde{Q}_i = Q_i - \bar{Q}_i$ deviation of the state h and Q around this steady state. We obtain then:

$$\begin{array}{rcl} \partial_t\tilde{h}_i & = & -B_i^{-1}\partial_x\tilde{Q}_i, \\ \partial_t\tilde{Q}_i & = & \zeta\partial_x\tilde{h}_i + \kappa\partial_x\tilde{Q}_i + \rho\tilde{h}_i + \phi\tilde{Q}_i, \end{array} \tag{5}$$

with appropriate ζ, κ, ρ and ϕ. Let us additionally define:

$$G = \begin{pmatrix} 0 & -B_i^{-1} \\ \zeta & \kappa \end{pmatrix}, \quad H = \begin{pmatrix} 0 & 0 \\ \rho & \phi \end{pmatrix}. \tag{6}$$

In the sub-critical regime (low flow speed), G has two eigenvalues satisfying $a_i = -\frac{\bar{Q}_i}{B_i\bar{h}_i} + \sqrt{g\bar{h}_i} > 0$ and $b_i = -\frac{\bar{Q}_i}{B_i\bar{h}_i} - \sqrt{g\bar{h}_i} < 0$. By applying the transformation

$$\begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix} = P^{-1}\begin{pmatrix} \tilde{h}_i \\ \tilde{Q}_i \end{pmatrix} \text{ with } P = \begin{pmatrix} 1 & 1 \\ -B_ia_i & -B_ib_i \end{pmatrix}, \tag{7}$$

we have a new system:

$$\partial_t\begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix} = \begin{pmatrix} a_i & 0 \\ 0 & b_i \end{pmatrix}\partial_x\begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix} + \begin{pmatrix} c_i & d_i \\ c_i & d_i \end{pmatrix}\begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix}, \tag{8}$$

with $\begin{pmatrix} c_i \\ d_i \end{pmatrix} = P^{-1}HP$.

The $N+1$ gate equations (3) can also be linearized and combined with the discharge conservation (4) to form a linear boundary condition as

$$\begin{pmatrix} \alpha_{i-1}(L,t) \\ \beta_i(0,t) \end{pmatrix} = \begin{pmatrix} m_i^{11} & m_i^{12} \\ m_i^{21} & m_i^{22} \end{pmatrix}\begin{pmatrix} \beta_{i-1}(L,t) \\ \alpha_i(0,t) \end{pmatrix} + \begin{pmatrix} b_i^1 \\ b_i^2 \end{pmatrix} g_i. \tag{9}$$

The first and the last junction have the same form:

$$\begin{array}{l} \beta_1(0,t) = m_1^{22}\alpha_1(0,t) + b_1^2g_1, \\ \alpha_N(L,t) = m_{N+1}^{11}\beta_N(L,t) + b_{N+1}^1g_{N+1}. \end{array} \tag{10}$$

where m_i^{jk}, $b_i^{1,2}$ are appropriate constants. In the sequel, for the sake of simplicity, we adopt the notation $\alpha_{i,0} = \alpha_i(0,t)$, $\alpha_{i,L} = \alpha_i(L,t)$ and similarly for β_i.

We add an integrator to inputs g_i as follows:

$$\dot{g}_i = u_i, \quad i = 1, ..., N+1 \tag{11}$$

In such a way, system (8)-(9) can be rewritten in the abstract form (see e.g. (Curtain & Zwart 1995), or (Pham et al. 2010, Pham et al. 2012)):

$$\begin{array}{rcl} \dot{z}(t) & = & \mathcal{A}z(t) + \mathcal{B}u(t), \quad t > 0 \\ z(0) & = & z^0, \end{array} \tag{12}$$

where $\mathcal{A}$ is the infinitesimal generator of a C_0-semigroup, $\mathcal{B}$ is a linear bounded operator. The new state z and the new control u are determined by

$$\begin{array}{rcl} z & = & \begin{pmatrix} g, \\ v - \mathbf{B}g \end{pmatrix}, \quad v = (\alpha_1 \cdots \alpha_N \quad \beta_1 \cdots \beta_N)^T \\ u & = & (u_1 \cdots u_{N+1})^T, \end{array} \tag{13}$$

with appropriate bounded operator $\mathbf{B}$ (see (Pham et al. 2012) for more detail). In this form, we can employ the C_0-semigroup theory (see (Curtain & Zwart 1995)) to establish the well-posedness as well as the existence of an optimal control for system (8)-(9).

3 MODEL-BASED PREDICTIVE CONTROL FOR IRRIGATION NETWORKS

Let us consider system (8)-(9) and recall for it the principle of MPC or Receding Horizon Optimal Control:

- At each time t, we obtain the current state $z(t)$.
- Then, for a given prediction time T and a cost function J, we compute the optimal solution of the problem:

$$\min_{\bar{u}\in\mathbf{L}_2(t,t+T;\mathbb{R}^{N+1})} J(z(t);\bar{u})$$
$$\text{s.t. } \dot{\bar{z}}(\tau) = \mathcal{A}\bar{z}(\tau) + \mathcal{B}\bar{u}(\tau),\ \bar{z}(t) = z(t),$$

where the notation $\bar{\cdot}$ stands for the predicted variables.

- The first part of the optimal control is applied on the system in period $[t, t+\sigma)$ for a small σ.
- The procedure restarts at $t+\sigma$.

One can note that since the actual state $z(t)$ is updated at each sampling step, the resulting control $u(t)$ is in fact in a feedback form which takes advantage of a receding horizon strategy in comparison to an open-loop optimal control.

We intend to employ this strategy to stabilize system (8)-(9) using the following optimization problem:

$$\begin{aligned}\min_u J &= \sum_{i=1}^{N+1}\int_0^T m_i(g_i,u_i)dt \\ &\quad + \sum_{i=1}^{N}\int_0^T\int_0^L l_i(\alpha_i,\beta_i)dxdt \\ &\quad + \sum_{i=1}^{N+1} m_i^f(g_i(T)) + \int_0^L l_i^f(\alpha_i(\cdot,T),\beta_i(\cdot,T))dx, \\ &\quad \text{s.t. (8)-(9)}\end{aligned} \tag{14}$$

The stage cost functions m_i and l_i and the terminal cost functions m_i^f and l_i^f are taken in quadratic form:

$$\begin{aligned} m_i(g_i,u_i) = q_i g_i^2 + r_i u_i^2,\ l_i(\alpha_i,\beta_i) = \begin{pmatrix}\alpha_i & \beta_i\end{pmatrix} Q_i \begin{pmatrix}\alpha_i \\ \beta_i\end{pmatrix}, \\ m_i^f(g_i) = q_i^f g_i(T)^2,\ l_i^f(\alpha_i,\beta_i) = \begin{pmatrix}\alpha_i & \beta_i\end{pmatrix} Q_i^f \begin{pmatrix}\alpha_i \\ \beta_i\end{pmatrix}, \end{aligned} \tag{15}$$

Using the transformation (13), the optimization problem (14) can be put in the following form:

$$\begin{aligned}\min_u J &= \int_0^T \langle z(t), Mz(t)\rangle dt + \langle u, Ru\rangle \\ &\quad + \langle z(T), M^f z(T)\rangle, \\ &\text{s.t. (12),}\end{aligned} \tag{16}$$

with appropriate definite positive operators M, R and M_f. In this form, we can show that there exists an optimal solution (see (Curtain & Zwart 1995)), which guarantees the feasibility at each sampling instant. In addition, we can choose the weighting parameters q_i, r_i, Q_i, q_i^f and Q_i^f (which inspired from the Lyapunov function proposed by (Coron, D'Andréa-Novel & Bastin 2007)) in order that the closed-loop system by MPC is asymptotically stable at the origin (see (Pham et al. 2010)).

4 DISTRIBUTED MODEL-BASED PREDICTIVE CONTROL

It is however difficult to solve the above optimization problem with a centralized control structure due to the computational complexity and to the robustness of the controller. In this section, we consider the prediction decomposition algorithm in order to get a distributed control scheme.

4.1 Prediction decomposition

Let us recall firstly the principle of this approach for a general optimization problem. Consider the following problem:

$$\min_{u,v} J(u,v) = \sum_{i=1}^{N} J_i(u_i,v_i), \tag{17}$$

$$\text{s.t. } \theta_i(u,v) = v_i - \sum_{j\neq i} H_{ij}(u_j,v_j) = 0,\ i=1,...,N \tag{18}$$

where $u=(u_1,...,u_N)$ is the decision variable, and $v=(v_1,...,v_N)$ is interaction variable. The term $H_{ij}(u_j,v_j)$ represents the influence of sub-system j to sub-system i.

In order to deal with the constraints, we use the augmented Lagrangian, which can be viewed as a mix of Lagrangian and penalty method:

$$L_c(u,v,p) = \sum_{i=1}^{N} J_i(u_i,v_i) + \langle p_i, \theta_i(u,v)\rangle + \tfrac{c}{2}\|\theta_i(u,v)\|^2$$

where c is a positive constant and p_i is the multiplier associated with constraint (18). The original constrained optimization problem is now equivalent to finding a saddle-point of $L_c(u,v,p)$. Thank to the quadratic term of the constraint, the convexity of the problem is enforced therefore, the convergence of dual algorithms (where we find alternatively $\min_{u,v} L_c(u,v,p)$ with a fixed p then $\max_p L_c(u,v,p)$ with (u,v) found in the previous step) is ensured (see (Cohen & Zhu 1984)).

By using linearization of the square of the constraint, (Cohen 1980) proposed different methods to decompose problem (18) into N sub-problems (each corresponds to control input u_i and can be solved by one agent). These approaches were applied in the context of distributed MPC e.g. by (Georges 2006) and (Rantzer 2009). The prediction decomposition algorithm consists of the following step:

1. At iteration $k=0$: choose p_i^0, $i=1,...,N$ and $u^0=(u_1^0,...,u_N^0)$ and $w^0=(w_1^0,...,w_N^0)$.
2. At iteration k: Each agent solves the following problem in (u_i,v_i):

$$\begin{aligned}&\min_{u_i,v_i} J_i(u_i,v_i) + \tfrac{1}{2\epsilon}\|u_i - u_i^k\|^2 \\ &+ \left\langle p_i^k + c\theta_i(u^k,v^k), \tfrac{\partial\theta_i}{\partial u_i}(u^k,v^k)u_i + \tfrac{\partial\theta_i}{\partial v_i}(u^k,v^k)v_i\right\rangle \\ &\text{s.t. } v_i = w_i^k\end{aligned}$$

(19)

Let u_i^{k+1}, v_i^{k+1} be a solution and μ_i^{k+1} the associated multiplier with constrain $v_i = w_i^k$.

3. Update p_i and w_i according to

$$\begin{array}{rcl} w_i^{k+1} & = & w_i^k - \epsilon(\mu_i^{k+1} + p_i^k), \\ p_i^{k+1} & = & p_i^k + \rho(\theta_i(u^k, v^k)) \end{array} \quad (20)$$

4. If $\|p^{k+1} - p^k\| + \|w^{k+1} - w^k\|$ is sufficiently small: stop, otherwise return to step 2 with k replaced by $k+1$

In this algorithm, the interaction variables v_i is fixed to its prediction value given by previous iteration. As consequence, problem (19) is in fact minimized only in terms of u_i, which reduces the number of decision variables for each sub-problem.

Note that in the above algorithm, it is not necessary to have a coordinator since information can be exchanged directly between agents.

4.2 Application to linearized Saint-Venant equations

We apply now the above approaches to problem (14) to decompose it into $N+1$ sub-problems. Let us first introduce the interconnection variables of each sub-system as:

$$\begin{array}{rcl} q_{i,+} & = & m_{i+1}^{12}\alpha_{i+1,0} + b_{i+1}^1 g_{i+1}, \\ q_{i,-} & = & m_i^{21}\beta_{i-1,L}, \end{array} \quad (21)$$

These variables $q_{i,+}$ and $q_{i,-}$ play the role of v_i in the general presentation in the previous section. The above relations can be seen as constraints for the optimization problem:

$$\begin{array}{rcl} \theta_{i,+} & = & q_{i,+} - m_{i+1}^{12}\alpha_{i+1,0} - b_{i+1}^1 g_{i+1} = 0, \\ \theta_{i,-} & = & q_{i,-} - m_i^{21}\beta_{i-1,L} = 0, \end{array}$$

In the sequel, the dependence of $\theta_{i,+}$ and $\theta_{i,-}$ to their arguments will be omitted for the sake of clarity. Let us denote $q_i = (q_{i,+} \quad q_{i,-})^T$, $\theta_i = (\theta_{i,+} \quad \theta_{i,-})^T$ and $p_i = (p_{i,+} \quad p_{i,-})^T$ the associated multiplier to θ_i. Then the boundary conditions (9) can be rewritten as:

$$\begin{array}{rcl} \alpha_i(L,t) & = & m_{i+1}^{11}\beta_i(L,t) + q_{i,+}, \\ \beta_i(0,t) & = & q_{i,-} + m_i^{22}\alpha_i(0,t) + b_i^2 g_i \end{array} \quad (22)$$

1. At iteration $k=0$: Choose p_i^0, u_i^0 and q_i^0. Simulate the sub-system i to get α_i^0, β_i^0 and g_i^0.

2. At iteration k: Solve in parallel

   For $i = 1, ..., N$

$$\begin{array}{l} \min\limits_{u_i} \int_0^T m_i(g_i, u_i)dt + \int_0^T \int_0^L l_i(\alpha_i, \beta_i)dxdt \\ + \int_0^T [\frac{1}{2\epsilon}(u_i - u_i^k)^2 + C_{3,i}^k\alpha_{i,0} + C_{4,i}^k\beta_{i,L} + C_{5,i}^k g_i]dt \\ + m_i^f(g_i(T), u_i(T)) \\ \text{s.t. (8), (22) and } q_i = w_i^k \end{array} \quad (23)$$

   with

$$\begin{array}{rcl} C_{1,i}^k & = & [p_{i,+}^k + c\theta_{i,+}(q_{i,+}^k, \alpha_{i+1,0}^k, g_{i+1}^k)] \\ C_{2,i}^k & = & [p_{i,-}^k + c\theta_{i,-}(q_{i,-}^k, \beta_{i-1,L}^k)], \\ C_{3,i}^k & = & -m_i^{12}[p_{i-1,+}^k + c\theta_{i-1,+}(q_{i-1,+}^k, \alpha_{i,0}^k, g_i^k)], \\ C_{4,i}^k & = & -m_{i+1}^{21}[p_{i+1,-}^k + c\theta_{i+1,-}(q_{i+1,-}^k, \beta_{i,L}^k)], \\ C_{5,i}^k & = & -b_i^1[p_{i-1,+}^k + c\theta_{i-1,+}(q_{i-1,+}^k, \alpha_{i,0}^k, g_i^k)], \end{array}$$

   For the last control input ($i = N+1$):

$$\begin{array}{l} \min\limits_{u_i} \int_0^T [m_i(g_i, u_i) + \frac{1}{2\epsilon}(u_i - u_i^k)^2 + C_{5,i}^k g_i]dt \\ \quad + m_i^f(g_i(T), u_i(T)), \\ \text{s.t. } \dot{g}_i = u_i \end{array} \quad (24)$$

   Let u_i^{k+1}, q_i^{k+1} a solution and α_i^{k+1}, β_i^{k+1}, g_i^{k+1} the associated trajectory.

3. Send $(\alpha_{i,0}^{k+1}, g_i^{k+1}, q_{i,-}^{k+1})$ to sub-system $(i-1)$ and send $(\beta_{i,L}^{k+1}, q_{i,+}^{k+1})$ to sub-system $(i+1)$.

4. Each agent updates the multiplier and interaction variables according to

$$\begin{array}{rcl} w_i^{k+1} & = & w_i^k - \epsilon(p_i^k + \mu_i^{k+1}), \\ p_i^{k+1} & = & p_i^k + \rho\theta_i^{k+1} \end{array} \quad (25)$$

   and constants $C_{j,i}^{k+1}$, $j = 1, ..., 5$ using received information from neighbors.

5. Stop if $\|p^{k+1} - p^k\|$ is below a desired threshold. Otherwise, make $k \leftarrow k+1$ and return to step 2.

where μ_i^{k+1} is the multiplier associated with the constraint $q_i = w_i^k$.

4.3 Convergence

Thanks to the convexity and the well-posedness of the global problem and of each sub-problem, with sufficiently small ϵ and ρ, the above decomposition schemes converge to global optimum of (14) (see (Cohen & Zhu 1984) and (Cohen 1980)).

5 MULTIAGENT SYSTEMS

The main specificity of the multiagent paradigm is its way to tackle the problem of subdivision and distribution of control using a collection of autonomous entities, interacting with each other in a common environment, possibly belonging to an organisation (Ferber 1999). In a MAS, each agent has a *local view* of the system, through its *perceptions*, and is responsible for the control of some part of the whole system, through its *effectors*. Agents can control physical devices, such as a gate or a pumping station in an irrigation network, or software components of the systems, such as

selecting relevant weather forecasts or modeling users' behavior. Some agents may also be responsible for a combination of both physical and logical components.

To obtain a global behavior which is: coherent, meets the system's objective, and matches the local and global requirements, agents have to interact with each other. These interactions may be designed according to problem-specific requirements or based on existing coordination strategies such as auction, negotiation, or norms. Inter-agent interactions may create and maintain dynamic *organisations* among agents. These organisations can, in turn, support coordination strategies.

Centralized control often give very good results in terms of optimality of given solutions and properties provability. However many real-world systems are too complex to be controlled using classical, centralized control methods. Examples of characteristics of a system which make it extremely difficult to control with a centralized approach include:

- Heterogeneity: when a system is composed of sub systems running at different time or space scales (e.g. a pumping station and a gate);
- Openness: when new components may be added to, or removed from, the system;
- Fault tolerance: in the sense of an ability to degrade gradually in case of failure of a sub-system.

In contrast, the multiagent paradigm take those characteristics into account explicitly, at design and deployment phases. Heterogeneity is addressed by encapsulating coherent sub-systems in dedicated agents and adapting their behaviour to the appropriate time/space scale. As the coupling between agents is captured by their interactions, adding, removing or rearranging components does not require modification of individual agent's behavior. This behavior must be designed from the beginning with these constraints in mind. This loose coupling existing between agents also improves the fault tolerance of a MAS. Since there is no one single central controlling point, there is no single component which is critical for the system as a whole. The fact that an agent cannot directly control another one ensure that dependencies between agents are managed through interactions These dependencies can thus be modified at runtime, for example in response to failure of an agent or other modifications of the system's structure.

5.1 A multiagent approach

One key component of a MAS is the design of agents' behaviors. Multiagent approaches help designers to integrate in agents the different models and strategies of such a system (organization models, interactions models, environment models, trust/reputation management, etc.). We use the DIAMOND (Jamont & Occello 2007) method specially dedicated to the design of real world embedded MAS.

Overview Four main stages may be distinguished within our embedded multiagent design approach. The *definition of needs* defines what the user needs and characterizes whole system global functionalities. The second stage is the *multiagent-oriented analysis* which consists in decomposing a problem into a multiagent solution. The third stage of the method starts with a *generic design* which aims to build the multiagent system without distinguishing hardware/software parts. Finally, the *implementation stage* consists in partitioning the system in a hardware part and a software part to produce the code and the hardware synthesis.

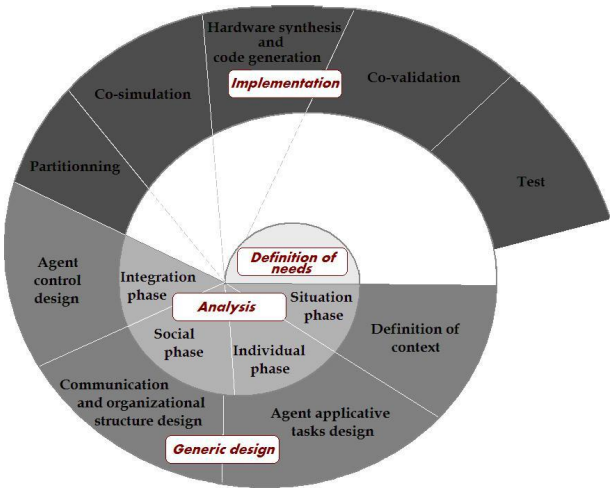


Figure 2 – The DIAMOND lifecycle

Multiagent-oriented analysis As the description of an agent gives a local vision (local behavior specification) of the system, the description of the society offers a global and external view of the system (external interaction and organization specifications). To obtain a complete view, we have to add an intermediate description concerning the social internal aspect. This description precises how individual behaviors interfere to compose the global behavior of the system and inversely how the global objectives have to be integrated to individual behaviors.

In the integrated approach proposed by DIAMOND, four phases exist to drive from a global characterization to the specification of local behaviors :

- the situation phase defines the overall setting, i.e., the environment, the agents with their main abilities and their contexts.
- an agent-centered phase defining individual

agents from an internal point of view (independently from social relations)

- a social phase describing interaction and organization from an external point of view
- a phase of socialization of individuals integrating social influences in the agent behaviors

In order to integrate all required functionalities into the agent's behavior, the designer must choose a decision-making architecture. This architecture is responsible to determine the next actions to be carried out by the agent's effectors, based on the current internal state of the agent and the perceived state of the environment. In this work we chose the ASTRO decision-making architecture, which is described briefly in the next section.

5.2 The ASTRO architecture

In the work described here each agent uses the ASTRO architecture (Occello et al. 1998), depicted by Fig. 3. This architecture combines the benefits from two classical classes of architectures: reactive and cognitive ones.

In reactive architectures, agent's actions are direct consequences of the perceived state of the environment. The agent has no internal model of the world or other kind of acquired knowledge. The behaviour of the agent is usually based on a set of predefined rules triggered by the agent's perceptions. An arbitration mechanism between these rules ensure the coherence of the behavior. In most reactive architectures there is no direct explicit communication between agents. This kind of architecture is particularly suited for low resources agents which must take actions quickly and do not need careful planning. They are however unable to carry out complex functionalities.

Cognitive architecture are based on an explicit world modeling process which builds an explicit representation of the world. This abstract construction is built from the perceptions of the agent and its previous states. Actions of the agent are based on a detailed analysis of the modeled state and potentially acquired knowledge. This kind of architecture is suited for agents which have to plan actions on the long run, acquire knowledge about their environment, or other forms of reasoning about the environment or themselves. However this kind of architecture often fails to work in highly constrained environments or when there is a need for fast decision making.

Hybrid architectures tend to combine the advantages of these two classes while avoiding their pitfalls.

In the ASTRO hybrid architecture, the integration of deliberative and reactive capabilities is possible through the use of parallelism in the structure of the agent. The central component of the architecture is a *world representation*. This representation is used to determine current goals and plans, making the architecture a cognitive one. In addition to this "cognitive" process, the world representation is continuously monitored by a *surveillance* process which is responsible for agent's reactive behavior. This behavior is based on triggers and guards which may be added or removed at runtime. When a trigger or a guard condition is met, its associated action is carried out without any planning or scheduling phase. This allows the agent to react quickly to situations requiring immediate actions, making the architecture reactive.

Using this architecture we combined an individual behavior, which consists in the control of an actuated gate using MPC, and a collective behavior responsible for the coordination of the individual behaviors to avoid local optima (using the decomposition-coordination presented in section 4).

In order to improve fault tolerance of the system, we added another collective behavior to the agents. This collective behavior detailed in the next section, gives the agents the ability to reconfigure themselves in order to adapt to some failure situations.

5.3 Adaptive failure response

In order to give adaptability to the controlling system as a whole, we added composition abilities to the agents. This ability allows agents that detect they are no longer able to control their gate to be integrated in still functioning agents.

The composition process unfolds as follows: first, the failing agent, wanting to initiate a composition, sends a composition request to the most suitable agent. This choice is based on the estimation of which composition will be able to fulfil its (now defunct) function. In our example, the most suitable agent is the upstream one. Second, upon composition acceptance from the upstream agent, the failing agent sends a topology reconfiguration message to its previously downstream agent. Third, the upstream and failing agents merge and the system resumes its controlling scheme, this new composed agent being responsible for the control of two pools.

Using this mechanism, the system can maintain its overall ability to control the water flow in the canal event though one agent lost its ability to act on the gate it controlled. Compared to a centralized control scheme or a static distributed MPC (DMPC), which would require to be modified if an actuator failure occurs, the presented control scheme is more adaptive.

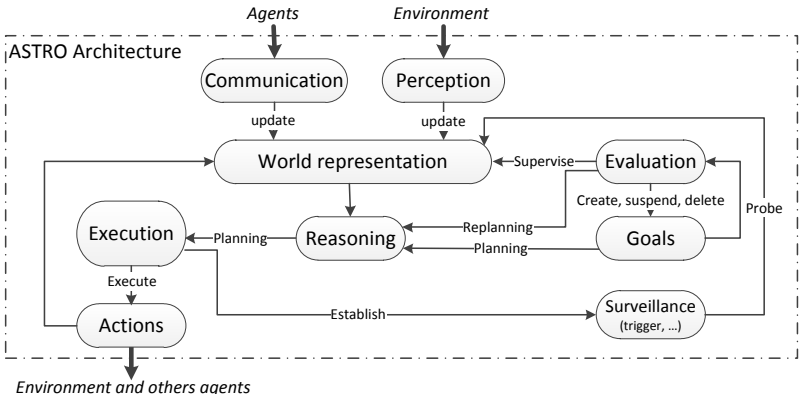


Figure 3 – ASTRO agent architecture

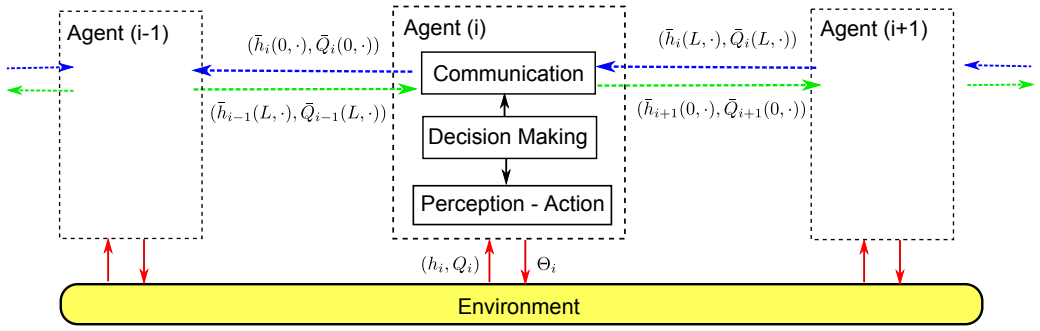


Figure 4 – Informations exchanged between agents

6 SIMULATION RESULTS

In this section, we present some simulation results carried out with a system of three cascaded pools of the same length $L = 3000m$ and width $B = 4.36m$. The slopes are $I_1 = 2.4 \times 10^{-4}$, $I_2 = 4.2 \times 10^{-4}$ and $I_3 = 6.2 \times 10^{-4}$. The steady state corresponds to $\bar{Q} = 4.1m^3/s$ and $\bar{h}_1 = 1.97m$, $\bar{h}_2 = 1.6m$ and $\bar{h}_3 = 1.4m$. The PDEs are solved with the Lattice Boltzmann Method (see (Pham et al. 2010)) with spatial step $\Delta x = 300m$ and $\Delta t = 1s$. The decomposition-coordination scheme uses $c = 7$, $\epsilon = 0.01$ and $\rho = 0.001$. The cost function is formulated with $T = 30s$.

Figure 5 presents the cost function of the decomposition-coordination scheme, in comparison with a centralized approach, with same turning parameters (which is the step size of the steepest descent method). We can notice that the our decomposition scheme converges faster than the centralized scheme. Nevertheless, the computation time on a Intel Core i7 3.4GHz, 8G RAM PC of the centralized scheme is around $1s$ whereas that of the prediction decomposition are $12.8s$ respectively. The reason is that agents have to realize several iterations before obtaining the optimal solution of sub-problem (19). The advantage of a distributed scheme in terms of computation time will be more evident when the number of subsystems increases.

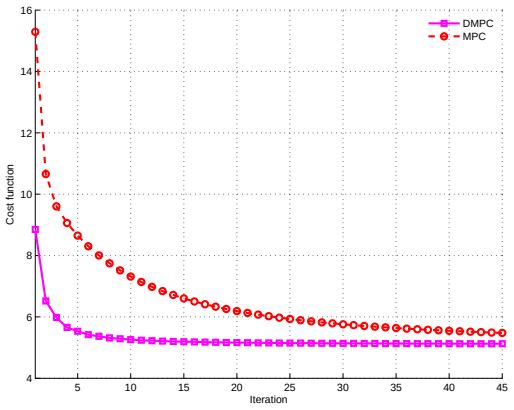


Figure 5 – Convergence of the decomposition-coordination scheme

We consider next the closed-loop system with MPC ($\sigma = 10s$). In order to reduce the communication cost (and the computation time), we limit the number of exchanges between agents to 30. This choice is justified by figure 5 where we can see that with 30 iterations, the decomposition schemes converge already to the optimum. The results are presented in figure 6. We can see that the physical variables h_i and Q_i converge to the steady state $\bar{h}_i$ and $\bar{Q}_i$.

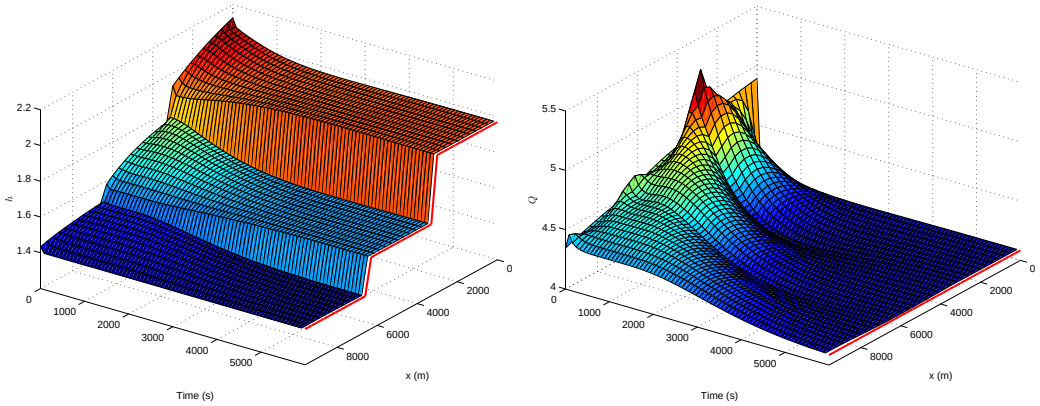


Figure 6 – Water level (left) and discharge (right) in the canal. The red lines represent the desired values

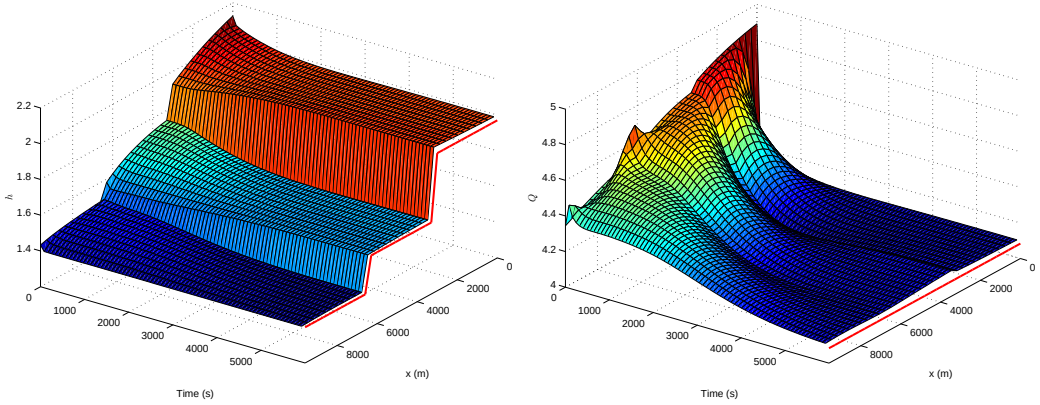


Figure 7 – Water level (left) and discharge (right) in the canal when the 2nd gate is blocked

In the second simulation we consider the case where the second gate is blocked. The result is presented in figure 7. We can see that water level and discharge are always regulated around the desired values despite of some small error.

7 CONCLUSIONS

In this paper we presented an application of the multiagent paradigm to the distributed, model-based predictive control of a water canal. In our experiments, each agent was able to control one of the canal's gates using MPC and inter agent communication in order to obtain an optimal distributed control. We simulated the failure of a gate's actuator and gave the system the ability to adapt to this failure by merging the appropriate agents into a new one. Combining the fine-grained, precise, control provided by DMPC on one hand and the strategic adaptation provided by MAS on the other hand, gave our approach an optimal control scheme in favorable cases and the ability to adapt to hardware failures by dynamically modifying the multiagent system's structure. The proposed agent composition mechanism maintains the global convergence verifiability. The presented system showed promising results regarding failure tolerance, ability to take global constraints into account and aptitude to scale up.

References

Bayen, A. M., Raffard, R. L. & Tomlin, C. J. (2006). Adjoint-based control of a new eulerian network model of air traffic flow, *IEEE transactions on Control systems technology* **14**(5): 804–818.

Carpentier, P. & Cohen, G. (1993). Applied mathematics in water supply network management, *Automatica* **29**(5): 1215–1250.

Christofides, P. D. & Daoutidis, P. (1997). Finite-dimensional control of parabolic PDE systems using approximate inertial manifolds, *Journal of Mathematica Analysis and Applications* **216**: 398–420.

Christofides, P., Scattolini, R. & de la Peña, D. M. (2013). Distributed model predictive control - a tutorial review and future research directions, *Computers and Chemical Engineering* **51**: 21–41.

Cohen, G. (1980). Auxiliary problem principle and decomposition of optimization problems, *Journal of Optimization Theory and Application* **32**(3).

Cohen, G. & Zhu, D. L. (1984). Decomposition coordination methods in large scale optimization problems: The nondifferentiable case and the use of augmented lagrangians, *Advances in Large Scale Systems* **1**: 203–266.

Colombo, R. M., Goatin, P. & Rosini, M. D. (2011). On the modeling and management of traffic, *ESAIM: Mathematical Modelling and Numerical Analysis* **45**: 853–872.

Coron, J., D'Andréa-Novel, B. & Bastin, G. (2007). A strict Lyapunov function for boundary control of hyperbolic systems of conservation laws, *IEEE transactions on automatic control* **52**: 2 – 11.

Curtain, R. & Zwart, H. (1995). *An Introduction to Infinite-Dimensional Linear Systems Theory*, Springer-Verlag.

Dubljevic, S., Mhaskar, P., El-Farra, N. H. & Christofides, P. D. (2005). Predictive control of transport-reaction processes, *Computer and Chemical engineering* **29**: 2335–2345.

Fawal, E., Georges, D. & Bornard, G. (1998). Optimal control of complex irrigation systems via decomposition-coordination and the use of augmented lagrangian, *Proceedings of the 1998 International Conference on Systems, Man and Cybernetics*, pp. 3874–3879.

Ferber, J. (1999). *Multi-Agent System: An Introduction to Distributed Artificial Intelligence*, Harlow: Addison Wesley Longman.

Findeisen, R., Imsland, L., Allgöwer, F. & Foss, B. A. (2003). State and output feedback nonlinear model predictive control: An overview, *European Journal of Control* **9**(3): 179–195.

Georges, D. (2006). Distributed model predictive control via decomposition-coordination techniques and the use of an augmented lagrangian, *Proceedings of IFAC Workshop on Nonlinear Model Predictive Control for Fast Systems*.

Georges, D. (2009). Infinite-dimensional nonlinear predictive control design for open-channel hydraulic systems, *Networks and Heterogeneous Media* **4**(2): 1–18.

Graf, W. H. & Altinakar, M. (2000). *Hydraulique fluviale - Écoulement et phénomènes de transport dans les canaux à géométrie simple*, Presses polytechniques et universitaires romandes.

Igreja, J., Lemos, J., Cadete, F., Rato, L. & Rijo, M. (2012). Control of a water delivery canal with cooperative distributed mpc, *Proceedings of American Control Conference*.

Ito, K. & Kunisch, K. (2002). Receding horizon optimal control for infinite dimensional systems, *ESAIM: Control, Optimisation and Calculus of Variations* **8**: 741–760.

Jamont, J.-P. & Occello, M. (2007). Designing embedded collective systems: The diamond multiagent method, *19th IEEE International Conference on Tools with Artificial Intelligence*, Vol. 2, IEEE Computer Society, pp. 91–94.

Liu, J., Chen, X., de la Peña, D. M. & Christofides, P. (2010). Sequential and iterative architectures for distributed model predictive control of nonlinear process systems, *AIChe J.* **56**: 2137–2149.

Negenborn, R., van Overloop, P., Keviczky, T. & Schutter, B. D. (2009). Distributed model predictive control of irrigation canals, *Networks and Heterogeneous Media* **4**(2): 359–380.

Occello, M., Demazeau, Y. & Baeijs, C. (1998). Designing organized agents for cooperation with real time constaints., *Collective Robotics, First International Workshop*, Vol. 1456, Springer, pp. 25–37.

Pham, V., Georges, D. & Besançon, G. (2010). Predictive control with guaranteed stability for hyperbolic systems of conservation laws, *Proceedings of 49th IEEE Conference on Decision and Control*.

Pham, V. T., Georges, D. & Besançon, G. (2012). Infinite-dimensional predictive control for hyperbolic systems, Revised version submitted to SIAM Journal of Control and Optimisation.

Rantzer, A. (2009). Dynamic dual decomposition for distributed control, *Proceedings of American Control Conference*.

Rawlings, J. B. & Mayne, D. Q. (2009). *Model Predictive Control: Theory and Design*, Nob Hill Pub, Llc.

Scattolini, R. (2009). Architectures for distributed and hierachical model predictive control, *Journal fo Process Control* pp. 723–731.

Serre, D. (1999). *Systems of Conservation law 1: Hyperbolicity, entropies, shock waves*, Cambrige University Press.

Stewart, B., Venkat, A., Rawlings, J., Wright, S. & Pannocchia, G. (2010). Cooperative distributed model predictive control, *System & Control Letters* .

Zarate-Florez, J., Molina, J. J. M., Besancon, G. & Faille, D. (2012). Decentralized-coordinated model predictive control for a hydro-power valley, *Mathematics and Computers in Simulation* **91**: 108–118.