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

Quantized nonlinear feedback design by a split dynamic programming approach

Péter Koltai and Oliver Junge

***Abstract*— We propose a split version of the optimality principle and an associated split optimal policy iteration in order to solve the dynamic programming problem for medium sized quantized nonlinear systems which consist of weakly coupled subsystems of small size. We show convergence of the scheme in special cases and present numerical experiments for discrete and continuous time systems.**

I. Introduction

In larger *networked control systems*, it is often possible to decompose the overall system into subsystems which are only loosely physically coupled to each other. A naive approach to design a controller for the entire system would be to neglect these couplings and to design controllers for the individual subsystems separately. Not surprisingly, even in rather simple situations the resulting closed loop system might not be stable, cf. [1]. On the other hand, designing a controller for the entire system can in general be rather hard (unless the system's model is of a special form).

It is therefore a common idea to ask in which way a subsystem is influenced by the other subsystems and how this influence can be properly incorporated into the design of the controllers for the individual subsystems. For instance, an ISS- and small-gain theory based approach for the construction of Lyapunov functions for networked systems has been proposed in [2], [3].

In this contribution we approach this question in the context of the dynamic programming principle. *Dynamic programming* gives an elegant way to construct a controller (even an approximately optimal one) based on a (non-linear) fixed point equation for the so-called optimal value function of the system, cf. [4], [5]. Once a suitable approximation of this function is available, an associated (approximately optimal) controller can readily be computed. The main obstacle for a general application of this approach is the *curse of dimension*, i.e. the fact that in general the computational effort in approximating the value function numerically scales exponentially in the dimension of the underlying phase and control space.

The main idea advocated in this paper is to decompose the original fixed point equation for the entire system into a system of n equations for the individual subsystems which are non-linearly coupled. We solve this system by a split policy iteration which cyclically iterates on the subsystem equations. The main advantage of this approach is that (a) in each iteration we only need to optimize over the controls for a single subsystem and (b) we can use different discretizations for the subsystem-copies of the value function. This allows to choose discretizations in a way which alleviates the curse of dimension, given suitable assumptions on how the subsystems influence each other (we will assume weak coupling below). Here, we discuss the convergence of this *split* optimal policy iteration scheme for systems with finite or quantized state and control spaces and for LQR systems, and show numerical experiments for several coupled processes. In these we use a quantized state space and the associated set-oriented discretization of the optimality principle as developed in [6], [7].

Alternating (cyclical) optimization over reduced variables for the computation of an optimal solution is used in many fields. It has been excessively studied (a) in optimization [8]; (b) in game theory [9]–[11], where best-response strategies are iterated to obtain a Nash equilibrium; and (c) in reinforcement learning [12], where the game-theoretical approach is applied with only approximate best-response strategies. In Proposition II.3 below we show convergence to a Nash equilibrium. Apart from this, the main contribution of this work is to connect the split representation with weak coupling in order to develop a framework which alleviates the curse of dimension in non-linear centralized feedback design.

The paper is structured as follows: In Section II we introduce non-linear optimal control problems and the corresponding optimality principle, then we propose a split form of the latter and an iterative algorithm for its solution. Section III exemplifies our approach for LQR problems, and states a convergence result. In order to arrive at a numerical method, we consider in Section IV a possible discretization scheme which naturally introduces a state quantization for the numerical realization of the iteration steps. Finally, Section V shows numerical examples.

II. Optimization-based centralized feedback design

A. Nonlinear optimal control

Suppose we have a discrete time system $f : \mathcal{X} \times \mathcal{U} \to \mathcal{X}$ with state space $\mathcal{X} \subset \mathbb{R}^{d_\mathcal{X}}$ and control set $\mathcal{U} \subset \mathbb{R}^{d_\mathcal{U}}$ (which we here assume to be finite for simplicity of presentation) and target set $\mathcal{T} \subset \mathcal{X}$, we would like to find a feedback law (also called policy) $\mu : \mathcal{X} \to \mathcal{U}$ such that the iteration $x(k+1) = f(x(k), \mu(x(k)))$ enters $\mathcal{T}$ after a finite number of steps. Since there may be many such laws, and in order to optimize performance, we follow an optimization-based approach. For this, define a cost function $c : \mathcal{X} \times \mathcal{U} \to \mathbb{R}_+$ with $c(x,u) = 0 \Leftrightarrow x \in \mathcal{T}$, and search for μ minimizing the

This work has been supported by the German Science Foundation (DFG) within the Priority Programme 1305.

P. Koltai and O. Junge are with the Dept. of Mathematics, Technische Universität München, 85748 Garching, Germany. {koltai,junge}@ma.tum.de

ISBN: 978-90-367-6321-9

accumulated costs

$$J(x,\mu) = \textstyle\sum_{k\geq 0} c\big(x(k),\mu(x(k))\big),$$

$$\text{where } x(k+1) = f\big(x(k),\mu(x(k))\big),\ x(0) = x.$$

Through Bellman's optimality principle (see [4]), the optimal feedback is strongly connected to the *optimal value function* $V(x) = \min_\mu J(x,\mu)$:

$$V(x) = \min_{u\in\mathcal{U}} \big\{c(x,u) + V(f(x,u))\big\},\ V\big|_{\mathcal{T}} \equiv 0 \quad (1)$$

$$\mu(x) = \arg\min_{u\in\mathcal{U}} \big\{c(x,u) + V(f(x,u))\big\}. \quad (2)$$

For states x that can not be steered into the target we set $V(x) = \infty$. The dynamic programming equation (1) enables the numerical approximation of the optimal value function and of the optimal feedback by value- and policy iteration, see [5]. A problem with all these approaches is that the curse of dimension inhibits their usage essentially for $\dim \mathcal{X} \geq 4$. Similarly, a global minimization with respect to u in (1) is also very expensive for $\dim \mathcal{U} \geq 4$, unless some structural properties, like convexity of the objective function can be established.

B. Weakly coupled systems

As mentioned above, the approximation of the optimal value function in high dimensions is not possible without further structural assumptions: we need some regularity which we can exploit. Motivated by the application scenario of networked control systems, where several, originally independent subsystems cooperate to achieve a common goal, we are searching for this regularity for *weakly coupled systems*, to be defined in the following.

Let the system consist of n subsystems, having states $x_i \in \mathcal{X}_i$, $i = 1,\dots,n$, and being affected by controls $u_i \in \mathcal{U}_i$; i.e. $\mathcal{X} = \bigotimes_{i=1}^n \mathcal{X}_i$ with $\mathcal{X} \ni x = (x_1,\dots,x_n)$, and $\mathcal{U} = \bigotimes_{i=1}^n \mathcal{U}_i$ with $\mathcal{U} \ni u = (u_1,\dots,u_n)$. Weak coupling means that the ratios $\|\partial_{x_j} f_i\|/\|\partial_{x_i} f_i\|$ and $\|\partial_{u_j} f_i\|/\|\partial_{u_i} f_i\|$ are small for $j \neq i$, where ∂_{x_i} and ∂_{u_i} denote the derivative with respect to x_i and u_i, respectively. The costs are actually assumed to be non-coupled, i.e. $c(x,u) = \sum_{i=1}^n c_i(x_i,u_i)$.

For non-coupled systems, i.e. $\partial_{x_j} f_i = 0$ and $\partial_{u_j} f_i = 0$ for $j \neq i$, one can see that the optimal value function is also non-coupled, i.e. $V(x) = \sum_{i=1}^n V_i(x_i)$, with V_i being the optimal value function of the i^{th} subsystem. Ideally, this form of the optimal value function would essentially be retained in a weakly coupled system, since then (assuming the $\mathcal{X}_i$ are all low dimensional) an approximation of the V_i would be possible. Unfortunately, even for LQR problems[1] in general we have to expect that the optimal value function cannot be approximated by one of a non-coupled system, cf. [1]. Numerical evidence supports this also in the nonlinear case.

Hence, our strategy in what follows is to relax the optimality, and to search for *some* Lyapunov function (not necessarily the optimal value function itself), and some feedback which satisfies prescribed regularity constraints (and is possibly optimal under these).

[1] In LQR problems the dynamics is affine linear in x and u, and the cost is the sum of an x-quadratic and a u-quadratic part, cf. Section III-A

C. Split dynamic programming equation

Let us pick one arbitrary $i \in \{1,\dots,n\}$ and assume that optimal feedback laws $\mu_j : \mathcal{X} \to \mathcal{U}_j$ for $j \neq i$ are known. Then, by explicitly including these into the Bellman equation, we can write

$$\begin{aligned} V(x) &= \min_{u\in\mathcal{U}} \big\{c(x,u) + V(f(x,u))\big\} \\ &= \min_{u_i\in\mathcal{U}_i} \big\{c[\hat{\mu}_i](x,u_i) + V(f[\hat{\mu}_i](x,u_i))\big\}, \end{aligned}$$

where $\hat{\mu}_i := (\dots,\mu_{i-1},\mu_{i+1},\dots)$, and

$$\begin{aligned} c[\hat{\mu}_i](x,u_i) &:= c(x,(u_i,\hat{\mu}_i(x))), \\ f[\hat{\mu}_i](x,u_i) &:= f(x,(u_i,\hat{\mu}_i(x))). \end{aligned}$$

Hence, the value function V and the feedback $\mu = (\mu_1,\dots,\mu_n)$ solving (1) and (2) also solve

$$\begin{aligned} W_i(x) &= \min_{u_i\in\mathcal{U}_i} \big\{c[\hat{\nu}_i](x,u_i) + W_i(f[\hat{\nu}_i](x,u_i))\big\} \\ W_i\big|_{\mathcal{T}} &\equiv 0 \\ \nu_i(x) &= \operatorname{argmin}_{u_i\in\mathcal{U}_i} \big\{c[\hat{\nu}_i](x,u_i) + W_i(f[\hat{\nu}_i](x,u_i))\big\} \end{aligned} \quad (3)$$

with $\nu = \mu$ and $W_i = V$ for each $i = 1,\dots,n$. Although the converse does not hold (i.e. in general a solution $(W_1,\dots,W_n)$ and $(\nu_1,\dots,\nu_n)$ of (3) does not yield a solution to (1) and (2)), the following theorem justifies the usage of the form (3).

Theorem II.1. *Assume $c(x,u) \geq \delta > 0$ for every $x \in \mathcal{X}\setminus\mathcal{T}$ and $u \in \mathcal{U}$. Then it holds for any solution $(W_1,\dots,W_n)$ and $\nu = (\nu_1,\dots,\nu_n)$ of the split Bellman equation (3) that there is a function W such that $W_i = W$ for every $i \in \{1,\dots,n\}$, and W is the value function of the policy ν, i.e. $W = J(\cdot,\nu)$. Moreover, if $W(x) < \infty$ then the feedback ν steers the trajectory starting in x into $\mathcal{T}$ in finitely many steps.*

Proof: Since W_i and ν solve (3), we have

$$\begin{aligned} W_i(x) &= \min_{u_i\in\mathcal{U}_i} \big\{c[\hat{\nu}_i](x,u_i) + W_i(f[\hat{\nu}_i](x,u_i))\big\} \\ &= c(x,\nu(x)) + W_i\big(f(x,\nu(x))\big). \end{aligned} \quad (4)$$

Let $x \in \mathcal{X}$ be arbitrary. Setting $x(0) = x$ and $x(k+1) = f\big(x(k),\nu(x(k))\big)$, we obtain by iterating (4) that for any $N \in \mathbb{N}$

$$W_i(x) = \sum_{k=0}^{N-1} c\big(x(k),\nu(x(k))\big) + W_i(x(N)) \quad (5)$$

holds. If $x(N) \in \mathcal{T}$ for some N, then $W_i(x(N)) = 0$. If $x(N)$ stays in $\mathcal{X}\setminus\mathcal{T}$ for all $N \in \mathbb{N}$, then $W_i(x) = \infty$ by (5) and the assumption on the cost function. In all cases, $W_i(x)$ is independent from i, this shows $W_i = W$ for a W with $W(x) \in [0,\infty]$. Again by (5) we have that $W(x) = J(x,\nu)$. The assumption $c(x,u) \geq \delta > 0$ for $x \notin \mathcal{T}$ shows that $W(x) \neq \infty$ implies that $x(N) \in \mathcal{T}$ for some finite N. □

Remark II.2. The assumption on the cost function can be relaxed to $c(x,u) > 0$ for $x \in \mathcal{X}\setminus\mathcal{T}$, and still asymptotic convergence to $\mathcal{T}$ would hold. Since it would require additional technicalities, a proof is omitted here.

We have thus split the original problem (1) into n nonlinearly coupled ones (3), with the advantage that now only minimizations with respect to $u_i \in \mathcal{U}_i$ occur, which we assume to be numerically tractable. We intend to solve (3) iteratively by cyclically solving the subproblems, see Algorithm II.1. As the criterion for termination we check whether the policies do not change any more (recall that we assumed $\mathcal{U}$ to be a finite set).

The idea of separability occurs in the dynamic programming literature too, usually with other forms of coupling; see e.g. [13]–[16].

Algorithm II.1 Split optimal policy iteration

Initialize: Set $\nu_i^{(0)} = \nu_i^0$, $i = 1, \dots, n$, where ν_i^0 are the feedback laws of the decoupled system.
for $k = 0, 1, \dots$ **do**
  for $i = 1, \dots, n$ **do**
    Compute $W_i^{(k+1)}$ by solving
    $W_i^{(k+1)}(x) =$
    $\min_{u_i \in \mathcal{U}_i} \left\{ c[\hat{\nu}_i^{(k)}](x, u_i) + W_i^{(k+1)}(f[\hat{\nu}_i^{(k)}](x, u_i)) \right\}$
    and set
    $\nu_i^{(k+1)}(x) =$
    $\arg\min_{u_i \in \mathcal{U}_i} \left\{ c[\hat{\nu}_i^{(k)}](x, u_i) + W_i^{(k+1)}(f[\hat{\nu}_i^{(k)}](x, u_i)) \right\}$
  end for
  if $\nu^{(k+1)} = \nu^{(k)}$ **then**
    return $\nu^{(k)}$ and $W_j^{(k)}$, $j = 1, \dots, n$
  end if
end for

Setting $\hat{\nu}_i^{(k)} = (\nu_1^{(k+1)}, \dots, \nu_{i-1}^{(k+1)}, \nu_{i+1}^{(k)}, \dots, \nu_n^{(k)})$, Algorithm II.1 is a Gauß–Seidel type iterative method.

Apart from the tractability of the minimization with respect to the controls, an other advantage concerns the numerical solution: the subproblems of (3) can be solved on different grids. The i^{th} subproblem can be solved on a grid which is fine in the directions of the i^{th} subsystem, and coarse in the other directions; cf. Section IV below. Thus, the curse of dimension is greatly reduced in the representation of the value function and of the feedback, as well as in the minimization with respect to the controls.

Before proceeding to the numerical realization of the algorithm, let us first analyze whether convergence can be expected. To this end we first consider the case of finite state and control sets, and then linear-quadratic regulator problems.

Proposition II.3. *Assume that $\mathcal{X}$ and $\mathcal{U}$ are finite sets. Then the value functions in the split optimal policy iteration II.1 converge to a fixed point W^*. More precisely, $W_i^{(k)} \to W^*$ as $k \to \infty$ for every $i = 1, \dots, n$. Moreover, the optimal policies $\nu^{(k)}$ can be chosen such that they converge too.*

Proof: Observe that the structure of the iteration is such that

$$W_i^{(k)} \geq W_{i+1}^{(k)} \geq \dots \geq W_n^{(k)} \geq W_1^{(k+1)} \geq \dots \geq W_i^{(k+1)}$$

for $k \geq 0$ and $i = 1, \dots, n-1$, where the inequalities are meant pointwise. Since also $W_i^{(k)} \geq 0$ for every i and k, the $W_i^{(k)}$ must converge pointwise to, say, W_i^*. The above inequalities also imply $W_i^* = W^*$ for every i. Since $\mathcal{X}$ and $\mathcal{U}$ are finite sets, the set of all possible policies is finite too, hence the $W_i^{(k)}$ can be at most finitely many different vectors of length $|\mathcal{X}|$. Hence, there is a $K \in \mathbb{N}$ such that $W_i^{(k)} = W^*$ for $k \geq K$. This shows that W^* is a fixed point of the iteration.

In particular, the optimal policies $\nu^{(k)}$ can be chosen such that they converge too. □

III. Split optimal policy iteration for linear-quadratic regulator problems

In order to get an idea about the convergence properties of the proposed algorithm, we first consider LQR problems. Certainly, for this class, even for large problems, there are efficient solvers readily available. As the reader might have noticed, the general LQR problem does not fit into the framework discussed above, since there is no target set considered. Nevertheless, it is highly suitable to exemplify the main mechanisms underlying the split optimal policy iteration, and the structure allows to make quantitative statements about its convergence speed.

A. Linear-quadratic regulator problems

Linear-quadratic regulator (LQR) problems arise in the context when the linear system

$$x(k+1) = \mathbf{A}x(k) + \mathbf{B}u(k) \tag{6}$$

shall be controlled to the origin in an optimal way. Here, $\mathbf{A} \in \mathbb{R}^{m \times m}$ denotes the system matrix and $\mathbf{B} \in \mathbb{R}^{m \times r}$ the control matrix. Further, let $\mathbf{Q} \in \mathbb{R}^{m \times m}$ and $\mathbf{R} \in \mathbb{R}^{r \times r}$ be symmetric positive definite matrices. The task is to find a sequence $(u(k))_{k \geq 0}$, generating a sequence $(x(k))_{k \geq 0}$ by (6), such that the accumulated costs

$$\sum_{k \geq 0} x(k)^{\mathrm{T}} \mathbf{Q} x(k) + u(k)^{\mathrm{T}} \mathbf{R} u(k)$$

are minimal. It turns out that the optimality principle is equivalent to the *discrete algebraic Riccati equation* [17]

$$\mathbf{P} = \mathbf{A}^{\mathrm{T}}\mathbf{P}\mathbf{A} - \mathbf{A}^{\mathrm{T}}\mathbf{P}\mathbf{B}\left(\mathbf{R} + \mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{B}\right)^{-1}\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{A} - \mathbf{Q}, \tag{7}$$

where the unique symmetric positive definite solution $\mathbf{P}$ of this equation yields the optimal value function $V(x) = x^{\mathrm{T}}\mathbf{P}x$. Moreover, the optimal feedback is given by

$$\mu(x) = \mathbf{F}^{\mathrm{opt}}x = -\left(\mathbf{R} + \mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{B}\right)^{-1}\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{A}x. \tag{8}$$

B. Two subsystems

In order to get a better intuition about how the split optimal policy iteration works we show an update step for the case of two subsystems. For this, let us partition the involved matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{Q}$, and $\mathbf{R}$ into blocks according to the subsystem decomposition, i.e. $\mathbf{A} = (\mathbf{A}_{ij})_{i,j=1}^2$, $\mathbf{B} = (\mathbf{B}_{ij})_{i,j=1}^2$, $\mathbf{Q} = \operatorname{diag}(\mathbf{Q}_1, \mathbf{Q}_2)$, and $\mathbf{R} = \operatorname{diag}(\mathbf{R}_1, \mathbf{R}_2)$. Assuming that some feedback matrix $\mathbf{F} = (\mathbf{F}_{ij})_{i,j=1}^2$ is given (partitioned the same way), we show how the feedback ν_1 of the first subsystem is updated by the split optimal policy iteration algorithm. First, note that $\nu_1(x) = \begin{pmatrix}\mathbf{F}_{11} & \mathbf{F}_{12}\end{pmatrix} x$ and $\nu_2(x) = \begin{pmatrix}\mathbf{F}_{21} & \mathbf{F}_{22}\end{pmatrix} x$. Since ν_2 is fixed during this

update step, we have to merge it into the matrices $\mathbf{A}$ and $\mathbf{Q}$. We obtain a new system matrix

$$\mathbf{A}^{(1)} = \mathbf{A} + \begin{pmatrix} \mathbf{B}_{12} \\ \mathbf{B}_{22} \end{pmatrix} \begin{pmatrix} \mathbf{F}_{21} & \mathbf{F}_{22} \end{pmatrix},$$

and a new state cost matrix

$$\mathbf{Q}^{(1)} = \mathbf{Q} + \begin{pmatrix} \mathbf{F}_{21} & \mathbf{F}_{22} \end{pmatrix}^{\mathrm{T}} \mathbf{R}_2 \begin{pmatrix} \mathbf{F}_{21} & \mathbf{F}_{22} \end{pmatrix}.$$

The control- and control cost matrices reduce to

$$\mathbf{B}^{(1)} = \begin{pmatrix} \mathbf{B}_{11} \\ \mathbf{B}_{21} \end{pmatrix}, \text{ and } \mathbf{R}^{(1)} = \mathbf{R}_1.$$

With our notation from before: $f[\hat{\nu}_1](x, u_1) = \mathbf{A}^{(1)}x + \mathbf{B}^{(1)}u_1$, $c[\hat{\nu}_1](x, u_1) = x^{\mathrm{T}}\mathbf{Q}^{(1)}x + u_1^{\mathrm{T}}\mathbf{R}^{(1)}u_1$. The update step reduces to solving an LQR problem with matrix quadruple $\mathbf{A}^{(1)}$, $\mathbf{B}^{(1)}$, $\mathbf{Q}^{(1)}$, and $\mathbf{R}^{(1)}$. A solution via (7) and (8) yields $\nu_1^{\mathrm{new}}(x) = \begin{pmatrix} \mathbf{F}_{11}^{\mathrm{new}} & \mathbf{F}_{12}^{\mathrm{new}} \end{pmatrix} x$.

Now, that the new feedback matrix

$$\mathbf{F}^{\mathrm{new}} = \begin{pmatrix} \mathbf{F}_{11}^{\mathrm{new}} & \mathbf{F}_{12}^{\mathrm{new}} \\ \mathbf{F}_{21} & \mathbf{F}_{22} \end{pmatrix}$$

is obtained, the iteration continues with the update of the 2^{nd} subsystem with $\mathbf{F}^{\mathrm{new}}$ replacing $\mathbf{F}$.

We close the discussion on the split optimal policy iteration for LQR problems by stating the following convergence result. Since the proof is more involved, and would go beyond the scope of this manuscript, we refer the interested reader to [18].

Theorem III.1. *Given an LQR problem with arbitrary number of subsystems and non-coupled cost function (i.e. block diagonal state- and control cost matrices $\mathbf{Q}$ and $\mathbf{R}$), let the split optimal policy iteration generate a sequence $\{\mathbf{F}^k\}_{k\geq 0}$ from some initial feedback matrix $\mathbf{F}^0$. Then, if $\mathbf{F}^0$ is such that the control pair $\left(\mathbf{A}^{(1)}, \mathbf{B}^{(1)}\right)$ arising in the first update step is controllable, then $(\mathbf{F}^k)_{k\geq 0}$ converges globally to the optimal feedback solution $\mathbf{F}^{\mathrm{opt}}$. The convergence is*

a) monotone in the sense that the corresponding value functions decrease monotonically; and
b) quadratic for continuous time systems, i.e. a local error ε decreases as $\mathcal{O}\left(\varepsilon^{2^k}\right)$ as k increases;
c) linear for discrete time systems, i.e. a local error ε decreases as $\mathcal{O}\left(\varrho^k\varepsilon\right)$ as k increases, and $\varrho \to 0$ for vanishing coupling strength.

Remark III.2. Note that convergence is faster the weaker the coupling between the subsystems. If this would carry over to non-linear systems as well, Algorithm II.1 would terminate in just a couple of steps. There is numerical evidence for the examples considered in section V which supports this belief.

IV. Discretization of the optimality principle

A. Construction of a robust non-linear controller

The split optimal policy iteration alone merely gives a partial improvement compared with trying to solve (2) directly. It removes the curse of dimension connected to the minimization with respect to the controls in the optimality principle at the cost of introducing n coupled problems instead of just one, but it does not deal with the curse of dimension in the state variable. The value functions W_i are still defined on the whole state space, and a discretization is necessary to compute approximate optimal value functions and policies. First, we discuss one possible discretization which we used, then in the next section we show how to apply it in combination with the split optimal policy iteration in order to alleviate the effects of the curse of dimension. The direct application of the approach presented in the following to a coupled problem is of course an alternative way to solve the optimal control problem; and this will be the reference method we compare our approach with.

We use the global non-linear robust feedback construction method for systems with quantized state space introduced in [7]. For this we assume an optimal control problem is given as in section II-A.

Let a partition $P = \{\mathcal{X}_1, \ldots, \mathcal{X}_\ell\}$ of $\mathcal{X}$ be given, i.e. $\mathcal{X} = \cup_{k=1}^{\ell} \mathcal{X}_i$ and $\mathcal{X}_i \cap \mathcal{X}_j = \emptyset$ whenever $i \neq j$. Assume that the target set is consistent with the partition, meaning that $\mathcal{T}$ is the union of some partition elements (also called *boxes*). Let $\varrho : \mathcal{X} \to P$ be the canonical projection of states on the partition, i.e. $\varrho(x) = \mathcal{X}_i$ if $x \in \mathcal{X}_i$. The idea of the feedback construction exploits the concept of a dynamic game, where the one player (the "controller") tries to minimize a given quantity (the accumulated costs along a trajectory), and the other player (the "perturber") tries to maximize the very same quantity. At the beginning of each step the system is in some state $x \in \mathcal{X}$. The controller may choose a control value $u \in \mathcal{U}$ with which the next step is carried out, and *then* the perturber decides from which state $y \in \varrho(x)$ the step is carried out. The state $f(y, u)$ is where the next step starts.

The optimal value function V_P of this game is uniquely characterized by the dynamic programming equation

$$V_P(x) = \min_{u\in\mathcal{U}} \sup_{y\in\varrho(x)} \{c(y, u) + V_P\left(f\left(y, u\right)\right)\}, \qquad (9)$$

with $V_P\big|_{\mathcal{T}} \equiv 0$, and the optimal policy is given by

$$\mu_P(x) = \arg\min_{u\in\mathcal{U}} \sup_{y\in\varrho(x)} \{c(y, u) + V_P\left(f\left(y, u\right)\right)\}. \qquad (10)$$

Since no matter where one starts in a given partition element $\mathcal{X}_i$, the perturber chooses the worst starting point $y \in \mathcal{X}_i$ anyway, one finds that V_P and μ_P are constant on partition elements. Hence, once these objects are computed, the implementation of the corresponding feedback controller is simple. For an analysis and details on computational issues and complexity we refer to [7], [19], [20]. Note that this construction yields a quantized feedback, since the only information it needs is the partition element $\mathcal{X}_i \ni x$.

B. Ignorant policies

As discussed in section II-B, theoretical and numerical evidence suggests that the optimal value function even of an only weakly coupled system does not have to possess a regular structure, e.g. of the type $V(x) \approx \sum_{i=1}^{n} V_i(x_i)$. To remedy this, we propose to drop the aim of computing the optimal value function itself, and try to compute *some* value function and corresponding feedback with some prescribed regularity properties. More precisely, observe that since the coupling of the subsystems is weak, the feedback ν_i of the i^{th} subsystem does not influence the evolution of, say, the j^{th}

subsystem ($j \neq i$) strongly. This suggests that there may exist *some* feedback ν such that the ν_i vary only slightly in the x_j coordinates.

We utilize this by solving the subproblems of the split optimal policy iteration on different partitions, namely such that the partition where the ν_i is computed on is only fine in the x_i coordinate, and coarse in every other, cf. Figure 1. This is why we call the resulting policy *ignorant*. Note that by construction the number of partition elements is much smaller than for a fully resolved partition. In a way, this *sparse* partition resembles the idea behind spare grids; see [21], [22].

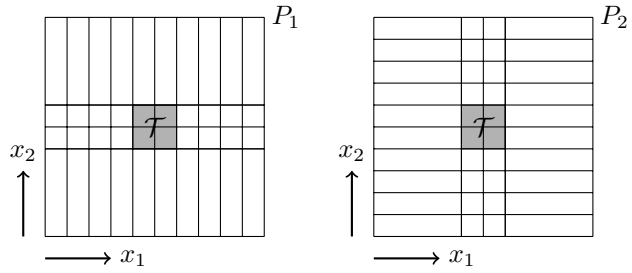

Fig. 1. Left: a possible partition P_1, which is fine in the x_1-direction and coarse in the others. Right: a possible partition P_2, which is fine in the x_2-direction, and coarse in the others. The gray rectangle in the center ($\mathcal{T}$) denotes the target set. The feedback law ν_1 is constant on elements of the P_1-partition, ν_2 is constant on elements of the P_2-partition.

Including the discretization on different partitions, Algorithm II.1 reads as Algorithm IV.1. Similarly to Theorem II.1 we can show the following stabilizing property of the computed feedback.

Theorem IV.1. *Assume $c(x,u) \geq \delta > 0$ for every $x \in \mathcal{X}\setminus\mathcal{T}$ and $u \in \mathcal{U}$. Then it holds for any fixed point $(W_1,\dots,W_n)$ and ν of Algorithm IV.1, and any $x \in \mathcal{X}$:*
(a) $J(x,\nu) \leq W_i(x)$ for all $i \in \{1,\dots,n\}$; and
(b) if $W_i(x) < \infty$ for an $i \in \{1,\dots,n\}$, then ν steers the system in finite steps into the target set.

Proof sketch: The proof essentially follows the same lines as the one of Theorem II.1. However, the value functions W_i are worst-case value functions under the action of the perturber. Since the original dynamics is one special case of the perturbed dynamics (namely, when the perturber always picks $x \in \rho_i(x)$), we obtain (a). Statement (b) follows with (a) and the assumption on the cost function, just as before. □

C. Homotopic strategy

Algorithm IV.1 is not expected to converge globally. Especially, for not carefully chosen initial policies the subproblems in Algorithm IV.1 are not controllable at all. To circumvent this, in addition to starting with initial policies equal to the optimal policies of the non-coupled problem, we also start the iteration with small coupling parameters and raise them during the iteration to the desired values. We call this a *homotopic strategy*. The advantages of this computational strategy are twofold:

a) As mentioned already, and as the numerical examples show, it enables the computation of a stabilizing feedback controller even for coupling strengths which are not necessarily weak; cf. section V, and e.g. Figure 2.
b) During the computation, one obtains feedback controllers for all parameter values from non-coupled to fully coupled along the homotopic range.

Algorithm IV.1 Split optimal policy iteration with discretization on different partitions

Given: n partitions $P_1,\dots,P_n$ of the whole state space $\mathcal{X}$, and associated canonical projections ϱ_i
Initialize: Set $\nu_{P_i}^{(0)} = \nu_{P_i}^0$, $i = 1,\dots,n$, where $\nu_{P_i}^0$ are feedback laws of the non-coupled system on the partition P_i
for $k = 0,1,\dots$ **do**
  for $i = 1,\dots,n$ **do**
    Compute $W_{P_i}^{(k+1)}$ by solving
    $W_{P_i}^{(k+1)}(x) =$

$$\min_{u_i \in \mathcal{U}_i} \sup_{y \in \varrho_i(x)} \left\{ c[\hat{\nu}_{P_i}^{(k)}](y,u_i) + W_{P_i}^{(k+1)}(f[\hat{\nu}_{P_i}^{(k)}](y,u_i)) \right\}$$

    on the partition P_i, and set
    $\nu_{P_i}^{(k+1)}(x) =$

$$\arg\min_{u_i \in \mathcal{U}_i} \sup_{y \in \varrho_i(x)} \left\{ c[\hat{\nu}_{P_i}^{(k)}](y,u_i) + W_{P_i}^{(k+1)}(f[\hat{\nu}_{P_i}^{(k)}](y,u_i)) \right\}$$

  end for
  if $\nu^{(k+1)} = \nu^{(k)}$ **then**
    return $\nu^{(k)}$ and $W_{P_j}^{(k)}$, $j = 1,\dots,n$
  end if
end for

V. Numerical examples

A. The thermofluid process

We consider a thermofluid process as described in [23]. The process involves two tanks (called TB and TS in the following) containing some fluid which' temperature and fill height shall be controlled. The coupling of the tanks is regulated by two valves (with opening ranging from 0 to 1), deciding how much fluid is pumped from one tank into the other. This defines the coupling, and is fixed during the process. Essentially, we have two coupled two dimensional continuous time subsystems, each with a two-dimensional control input.

First, we fix the fill heights and focus on the control of temperatures ϑ_{TB} and ϑ_{TS}. This leaves us with two (coupled) one dimensional subsystems, each having one dimensional controls from $\mathcal{U}_1 = \mathcal{U}_2 = [0,1]$. The overall system is now linear, this will allow for a demonstration of the split optimal policy iteration method with the discrete computation of ignorant policies (section IV-B), and of the theoretical results in the LQR case as well (section III-B). We chose coupling constants 0.19 and 0.22.

The operating point to stabilize is given by $\bar{\vartheta}_{TB} = 294.7$ and $\bar{\vartheta}_{TS} = 300.2$ (in K). The operating point stays invariant by applying the constant controls $\bar{u}_1 = \bar{u}_2 = 0.5$. Introducing new variables $\theta = (\theta_{TB},\theta_{TS}) := (\vartheta_{TB} - \bar{\vartheta}_{TB}, \vartheta_{TS} - \bar{\vartheta}_{TS})$ and $v := u - \bar{u}$, the system

evolution is given by $\dot{\theta}(t) = \mathbf{A}\theta(t) + \mathbf{B}v(t)$, with

$$\mathbf{A} = 10^{-3}\begin{pmatrix} -8.6 & 5.4 \\ 5.6 & -5.6 \end{pmatrix}, \qquad \mathbf{B} = 10^{-2}\begin{pmatrix} -3.9 & 0 \\ 0 & 3.5 \end{pmatrix},$$

with $\dot{\theta}$ denoting the time derivative of $\theta(t)$.

1) Linear analysis: Since our analysis is based on discrete time systems, we consider a time-sampled version of the system, $\theta(k+1) = \mathbf{A}^\tau\theta(k) + \mathbf{B}^\tau v(k)$, for a sampling time $\tau > 0$, where

$$\mathbf{A}^\tau = e^{\mathbf{A}\tau}, \qquad \mathbf{B}^\tau = \int_0^\tau e^{\mathbf{A}(\tau-s)}\, ds\ \mathbf{B}.$$

We choose $\tau = 400$. Then, we apply the split optimal policy iteration as described in Section III-B for the discrete time LQR problem defined by $\mathbf{A}^\tau$, $\mathbf{B}^\tau$, and $\mathbf{Q}_1 = \mathbf{Q}_2 = 4 \cdot 10^{-4}$, $\mathbf{R}_1 = \mathbf{R}_2 = 1$. We observe fast convergence as predicted by Theorem III.1. In fact, we observed that the convergence speed depends on the sampling time τ, such that the smaller the sampling time, the faster the convergence to the respective optimal feedback matrix. Figure 2 (left) shows the trajectories of the optimally controlled system (for $\tau = 400$).

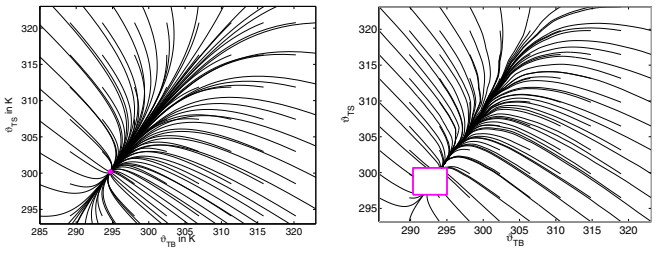

Fig. 2. Left: trajectories of the system simulated with the optimal policy of the corresponding LQR problem. Between two discrete sample points the trajectory of the continuous time evolution is drawn to show the dynamical behavior. The dot represents the operating point. Right: trajectories of the system simulated with the ignorant policies obtained from Algorithm IV.1.

2) Non-linear analysis: As one could expect, the optimal controller of the LQR problem does not respect the control restrictions $u_{1,2} \in [0, 1]$. For the non-linear analysis, as described in Section IV-B, the control sets $\mathcal{U}_1$ and $\mathcal{U}_2$ are finite sets with 41 equispaced controls between 0 and 1. The partitions P_1 and P_2 are uniform 64×16 and 16×64 partitions of $\mathcal{X} = [285, 323] \times [294, 323]$, respectively, and the target set is chosen to be the union of several boxes on both partitions, such that it encloses the operating point. The sampling time $\tau = 400$ is used, just as before.

Remark V.1. Due to the discretization we use, much smaller values of τ are not applicable, because the controllability of the perturbed system relies on the fact that there is a control such that the image of each partition element is disjoint with the partition element itself. Other wise the perturber could keep the system in the very same partition element for all times. This limits the sampling time from below, since a minimum "mobility" has to be guaranteed.

We carried out Algorithm IV.1 with the homotopic strategy to obtain W_1, W_2, ν_1, and ν_2. We raised the coupling parameters from zero to their final values in 10 equispaced steps, and performed 3 cycles of the split optimal policy iteration in each step. We did not obtain convergence for each coupling parameter value after the 3 cycles. However, the changes between the last iterates were small, indicating that a fixed point is close and it is safe to proceed to the next parameter value. Figure 3 shows the resulting value functions W_1, W_2. The rectangles show the target set. Figure 2 (right) shows a simulation of several trajectories steered with the policy $\nu = (\nu_1, \nu_2)$.

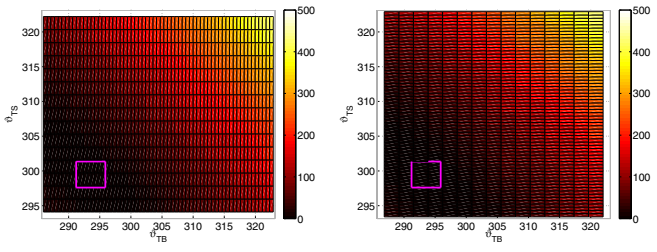

Fig. 3. Left: The value function W_1 of the 1^{st} subsystem on the partition P_1 computed by the split optimal policy iteration Algorithm IV.1. Right: The value function W_2 of the 2^{nd} subsystem on the partition P_2 computed by the split optimal policy iteration Algorithm IV.1.

The main numerical effort during the computation—either by the direct method from section IV-A, or by our new one described in Algorithm IV.1—is dominated by the mapping of sample points. This number is $|P| \cdot |\mathcal{U}| \cdot \#\{\text{sample points}\}$.

The numerical effort for Algorithm IV.1 amounted to $1.0 \cdot 10^7$ map evaluations, while the effort for the reference method for a uniform 32×32 partition would have been $6.9 \cdot 10^6$ evaluations. Clearly, the advantage of the split optimal policy iteration—e.g. that it scales linearly with the control space dimension, while the standard methods scales exponentially—is not visible in this 2d example, however it will get prominent in higher dimensions, as shown in the next examples.

B. The 4d thermofluid process

Having analyzed the reduced process, we now apply our approach for the full thermofluid process. Since the mathematical model is complicated, we refer to [23] for a full description. The process is given by a system of 4 coupled non-linear differential equations, describing the evolution of the temperature $(\vartheta_{TB}, \vartheta_{TS})$ and fill height (l_{TB}, l_{TS}) in the two tanks, influenced by 2 control parameters per tank. It is natural to consider the tanks as subsystems, giving us two subsystems with 2d state and 2d control spaces each. The coupling constants are chosen 0.19 and 0.29.

In order to obtain a monotone convergence due to Proposition II.3, Algorithm IV.1 is run on identical partitions P_1 and P_2, both being a $16 \times 16 \times 16 \times 16$ uniform partition of the state space $[0.26, 0.4] \times [285, 323] \times [0.26, 0.4] \times [285, 323]$. The control set is uniformly discretized into $41 \times 11 \times 21 \times 11$ controls, i.e. subsystem TB has 451, and subsystem TS has 231 control inputs. The target set is chosen to be $\mathcal{T} = [0.33, 0.35] \times [292, 297] \times [0.33, 0.35] \times [297, 300]$, and the sampling time $\tau = 400$ is used. A quadratic cost function analog to the 2d case is used to obtain an optimal control problem.

A direct application of the method from section IV-A

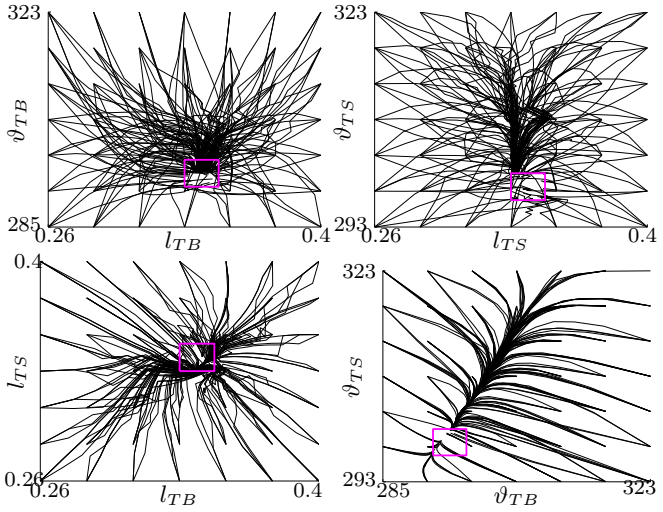

Fig. 4. Trajectories of the 4d thermofluid process steered with the optimal control, projected onto different coordinates. Top left: (l_{BT}, ϑ_{TB}). Top right: (l_{TS}, ϑ_{TS}). Bottom left: (l_{TB}, l_{TS}). Bottom right: $(\vartheta_{TB}, \vartheta_{TS})$.

would require $6.8 \cdot 10^9$ evaluations of the flow map of the differential equation. Instead, we apply our algorithm with homotopic strategy, using 10 homotopic parameter values, and allowing for a maximum of 3 cycles for each value. The associated split optimal policy iterations converges, and $1.3 \cdot 10^9$ evaluations of the flow map were used.

Throughout the iteration, for every parameter value considered, more than 99.9 % of the boxes were controllable. To understand why the small portion of boxes fails to be controllable, note that the fill heights l_i, $i \in \{TB, TS\}$, evolve on average 100 times faster than the temperatures ϑ_i. Still, the sampling time is chosen such that even the temperatures can satisfy the mobility requirement mentioned before. This, combined with the finite resolution of the control space, may cause that the trajectories miss the target. This difficulty is omnipresent in sampled-time control of systems exhibiting multiscale behavior.

Figure 4 shows trajectories simulated with the obtained optimal policy, projected onto different coordinates. Note that the coupling is not weak at all: trajectories which start with the same TB-coordinates follow different paths in the TB-projection; and the same holds for TS too.

It is interesting to observe that that the trajectories in the temperatures-plane in Figure 4 look like those in Figure 2. This can be understood by recalling the multiscale nature of the process. The temperatures are slow compared with the fill heights, thus they dominate the accumulated costs. If they would deviate from the optimal trajectories computed for the reduced 2d process, they would accumulate more costs. Thus, the similarity of the trajectory plots indicates that our method really computes a (nearly) optimal feedback.

C. A 6d and a 4d Kot–Schaffer growth-dispersal model

As our final example we consider a genuinely discrete time system. Let $p_i(k)$ denote the population of some species at location $i \in \{1, \dots, n\}$ and time k (one time unit could model here the life cycle of the species). Without interaction, the population at each location would evolve according to the rule $p_i(k+1) = g(p_i(k))$, for some nonlinear map g. However, the species is assumed to disperse spatially during its life cycles, resulting in the population dynamics [24]

$$p_i(k+1) = \sum_{j=1}^{n} w_{ij} g(p_j(k)), \tag{11}$$

with $\mathbf{W} := (w_{ij})_{i,j=1}^n$ being some coupling matrix. We assume that the self-dynamics of the population is given by the logistic-type map $g(p) = g(p; c) = rp \max\left\{1 - \frac{p}{c}, \varepsilon\right\}$, where $r > 0$ is the reproduction rate, $c > 0$ the carrying capacity of the environment, and $0 < \varepsilon \ll c$ the constant to which the population decreases if it exceeds the capacity c.

Our aim is to control the populations at each location i to a given value, by manipulating the carrying capacity of that location. Thus we obtain a nonlinear control system

$$f(p, u) = \sum_{j=1}^{n} w_{ij} g(p_j; u_j).$$

We consider the cost function $c(p, u) := \sum_{i=1}^n (p_i - p_i^*)^2 + (u_i - u_i^*)^2$, where p_i^* is the center point of the subsystem target set $\mathcal{T}_i$, and u_i^* is the control such that $f(p^*, u^*) = p^*$.

1) Population splitting: We take $n = 6$ and think of the locations of the species being spread along a line. Our aim is to control the leftmost and rightmost populations to their maximal value and keep the populations at the middle locations as small as possible. For this, let $r = 3$, $\varepsilon = 10^{-2}$,

$$\mathbf{W} = \begin{pmatrix} 0.92 & 0.03 & 0.02 & 0 & 0 & 0 \\ 0.05 & 0.9 & 0.07 & 0.02 & 0.02 & 0 \\ 0.03 & 0.03 & 0.82 & 0.07 & 0.02 & 0.02 \\ 0 & 0.02 & 0.07 & 0.82 & 0.03 & 0.03 \\ 0 & 0.02 & 0.02 & 0.07 & 0.9 & 0.05 \\ 0 & 0 & 0 & 0.02 & 0.03 & 0.95 \end{pmatrix}.$$

$\mathcal{U}_i = [0, 1]$ (discretized into 29 equispaced values), $\mathcal{T}_i = [0.56, 0.75]$ for $i = 1, 6$ and $\mathcal{T}_i = [0, 0.18]$ for $i = 2, 3, 4, 5$. The reason for the choice of the target sets is that with the maximal capacity 1 the maximal population is 0.75. We further assume that the population can be observed at every 7^{th} cycle, so we essentially work with the map f^7.

We apply Algorithm IV.1 for partitions P_i, $i = 1, \dots, 6$, which have 16 boxes in the i^{th} coordinate and 4 in all others (i.e. every partition contains 16384 boxes), use a homotopic strategy to change $\mathbf{W}$ in 8 steps from the identity matrix $\mathbf{I}$ to the one above, allowing at most 3 cycles of the split optimal policy iteration for each coupling scenario. The algorithm converges to a fixed point feedback controller, which steers any initial condition to the target set, in only 16 cycles. During the process the map f has been evaluated $5.1 \cdot 10^9$ times. If we would have carried out the computation with the reference method on a $4 \times \ldots \times 4$ partition, the number of f-evaluations would have been $1.1 \cdot 10^{15}$.

2) Higher reproduction rates: A single logistic map has a stable fixed point for $r \leq 3$, which splits into a periodic orbit with increasing periods of length 2^k ($k \to \infty$) as r increases approximately to 3.57. From that value on, the behavior is mostly chaotic [25]. We attempt to perform the above population splitting for $r = 3.5$. For this, let $n = 4$ (since

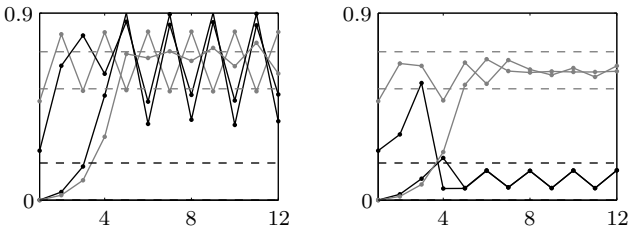

Fig. 5. Trajectories for the 4d Kot–Schaffer model, $r = 3.5$. Left: non-controlled trajectory. Right: controlled trajectory. The dashed black/gray lines indicate the target sets where the black/gray subsystems should be steered to.

the more complicated dynamics calls for a higher resolution of the state space, we decrease the number of subsystems),

$$\mathbf{W} = \begin{pmatrix} 0.90 & 0.02 & 0.01 & 0.02 \\ 0.05 & 0.95 & 0.02 & 0.03 \\ 0.03 & 0.02 & 0.95 & 0.05 \\ 0.02 & 0.01 & 0.02 & 0.90 \end{pmatrix}.$$

$\mathcal{U}_i = [0, 1]$ (discretized into 20 equispaced values), $\mathcal{T}_i = [0.54, 0.72]$ for $i = 1, 4$ and $\mathcal{T}_i = [0, 0.17]$ for $i = 2, 3$. The population is now observed in every single life cycle.

With these parameters, a single logistic map needs the state space to be partitioned into at least 16 elements (we use powers of two for algorithmic reasons) in order to achieve controllability. If we were to apply the reference method on a $16 \times 16 \times 16 \times 16$ discretization for this problem, $1.7 \cdot 10^{11}$ f-evaluations would be needed.

Instead, we apply Algorithm IV.1 for partitions P_i, $i = 1, \dots, 4$, which have 16 boxes in the i^{th} coordinate and 8 in all others; i.e. every partition contains 8192 boxes. Note that for each partition P_i the minimal resolution, 16, necessary for controllability is only needed in the i^{th} coordinate direction, thus having ν_i sufficiently resolved in the i^{th} coordinate, but not wasting resolution in others. We use a homotopic strategy to change $\mathbf{W}$ in 10 steps from the identity matrix $\mathbf{I}$ to the one above, allowing at most 3 cycles of the split optimal policy iteration for each parameter value. This gives a total of $3.1 \cdot 10^8$ f-evaluations. For the fixed point of the iteration holds $W_{P_i} < \infty$ on 8184, 8184, 8171 and 8177 from 8192 boxes for $i = 1, 2, 3, 4$, respectively. Theorem IV.1 tells us that for any $x \in \mathcal{X}$ such that $W_{P_i}(x) < \infty$ for at least one i the corresponding controller steers x into the target set. The handful of boxes which can not be controlled have low population values in every subsystem, and fail to satisfy the mobility requirement; cf. Remark V.1.

VI. Conclusions

We have introduced a split Bellman equation for coupled optimal control problems and proposed an iterative procedure—with complexity scaling linearly in the dimension of the control set—for its solution. Theoretical findings and numerical evidence suggests that the iteration converges very rapidly. The splitting allows for solving the subproblems on different grids; which alleviates the curse of dimension in the state variable too. Our numerical studies show a performance increase of a factor $10^1 - 10^5$, depending on the dimension; and also some difficulties if the process is slow in given regions of the state space (cf. "mobility"), or is multiscale. We would like to tackle both these problems in future work by using *event-based* control approaches. Further studies shall look at processes with even more systems and *sparse coupling structure*.

References

[1] P. Koltai and O. Junge, "Optimal value functions for weakly coupled systems: a posteriori estimates," *ZAMM*, vol. 94, pp. 345–355, 2014.
[2] S. N. Dashkovskiy, B. S. Rüffer, and F. R. Wirth, "Small gain theorems for large scale systems and construction of ISS Lyapunov functions," *SIAM J. Control Optim.*, 2009.
[3] L. Grüne and M. Sigurani, "Numerical ISS controller design via a dynamic game approach," proceedings of the 52nd IEEE Conference on Decision and Control - CDC 2013, to appear.
[4] R. Bellman, *Dynamic Programming*, 1st ed. Princeton University Press, 1957.
[5] D. P. Bertsekas, *Dynamic Programming and Optimal Control*. Athena Scientific, Belmont, Massachusetts, 1995, vol. II.
[6] O. Junge and H. M. Osinga, "A set oriented approach to global optimal control," *ESAIM Control Optim. Calc. Var.*, vol. 10, no. 2, pp. 259–270 (electronic), 2004.
[7] L. Grüne and O. Junge, "Global optimal control of perturbed systems," *J. Optim. Theory Appl.*, vol. 136, no. 3, pp. 411–429, 2008.
[8] J. Bezdek and R. Hathaway, "Some notes on alternating optimization," in *Lecture Notes in Computer Science*, 2002, vol. 2275, pp. 288–300.
[9] J. Robinson, "An iterative method of solving a game," *Annals of Mathematics*, vol. 54, pp. 296–301, 1951.
[10] D. Monderer and L. S. Shapley, "Potential games," *Games and Economic Behavior*, vol. 14, pp. 124–143, 1996.
[11] ——, "Fictitious play property for games with identical interests," *Journal of Economic Theory*, vol. 68, pp. 258–265, 1996.
[12] M. Littman, "Value-function reinforcement learning in Markov games," *Journal of Cognitive Systems Research*, vol. 2, pp. 55–66, 2001.
[13] D. P. Bertsekas, "Separable dynamic programming and approximate decomposition methods," in *IEEE T. Automat. Contr.*, vol. 52, no. 5, 2007, pp. 911–916.
[14] N. Meuleau, M. Hauskrecht, K. eung Kim, L. Peshkin, L. P. Kaelbling, T. Dean, and C. Boutilier, "Solving very large weakly coupled Markov decision processes," in *In Proceedings of the Fifteenth National Conference on Artificial Intelligence*, 1998, pp. 165–172.
[15] M. Lauer and M. Riedmiller, "An algorithm for distributed reinforcement learning in cooperative multi-agent systems," in *In Proceedings of the Seventeenth International Conference on Machine Learning*. Morgan Kaufmann, 2000, pp. 535–542.
[16] C. Guestrin, D. Koller, and R. Parr, "Multiagent planning with factored MDPs," in *In NIPS-14*. The MIT Press, 2001, pp. 1523–1530.
[17] E. D. Sontag, *Mathematical Control Theory: Deterministic Finite Dimensional Systems*, 2nd ed. Springer, New York, 1998.
[18] P. Koltai, "Split optimal policy iteration for LQR problems," e-print, arXiv:1404.5209.
[19] L. Grüne and O. Junge, "A set oriented approach to optimal feedback stabilization," *Systems and Control Letters*, vol. 54, no. 2, pp. 169–180, 2005.
[20] ——, "Approximately optimal nonlinear stabilization with preservation of the Lyapunov function property," in *Proceedings of the 46th IEEE Conference on Decision and Control*, 2007, pp. 702–707.
[21] C. Zenger, "Sparse grids," in *Parallel algorithms for partial differential equations (Kiel, 1990)*, ser. Notes Numer. Fluid Mech. Braunschweig: Vieweg, 1991, vol. 31, pp. 241–251.
[22] H.-J. Bungartz and M. Griebel, "Sparse grids," *Acta Numerica*, vol. 13, pp. 1–123, 2004.
[23] C. Stöcker and C. Schymura, "Verfahrenstechnischer Demonstrations- und Benchmarkprozess," 2011, http://spp-1305.atp.ruhr-uni-bochum.de/index.php?site=benchmark2.
[24] M. Kot and W. M. Schaffer, "Discrete-time growth-dispersal models," *Math. Biosci.*, vol. 80, pp. 109–136, 1986.
[25] W. de Melo and S. van Strien, *One-dimensional dynamics*. Springer-Verlag, 1993.