Abstract—A classical approach to design controllers for interconnected systems is to assume that the different subsystems operate at different time scales, then design simpler controllers within each time scale, and finally, certify stability of the interconnected system via singular perturbation analysis. In this work, we propose an alternative approach that also allows to design the controllers of the individual subsystems separately. However, instead of requiring a sufficiently large time-scale separation, our approach consists of adding a feed-forward term to modify the dynamics of faster systems in order to anticipate the dynamics of slower ones. We present several examples in bilevel optimization and cascade control design, where our approach improves the performance of currently available methods.

Index Terms—Bilevel optimization, cascade control, interconnected systems, nonlinear control design, singular perturbation, time-scale separation.

I. INTRODUCTION

INTERCONNECTED and nested systems are ubiquitous in control applications, but they may be challenging to analyze and design. If interconnected systems are composed by subsystems operating on multiple time scales [1] and a normal hyperbolicity condition holds [2], then each time scale can be studied independently, substituting dynamics of faster time scales by algebraic equations [3]. Such systems appear in engineering applications like power systems [4], [5], biological systems [6], motion control [7], electrical drives [8], etc. In that context, time-scale separation arguments, like singular perturbation analysis [9], [10], allow to certify when the stability guarantees derived in each separate time scale are preserved in the interconnected system. Standard singular perturbation considers only two time scales [11], although it can be extended to multiple ones [2], [5], [12]. Besides analysis, singular perturbation is also a powerful tool for control design [13], e.g., as a model-reduction technique [14]: Complex systems on a single time scale can be artificially separated into subsystems on different time scales and, thus, simplify their analysis and controller design. Singular perturbation analysis can then provide additional conditions, e.g., on the control parameters [5], to ensure that the interconnected system remains stable. Some examples of these applications are hierarchical control architectures, like cascade control [15], or iterative optimization algorithms, like dual ascent [16], and interior point methods [17]. However, for more than two time scales, such singular perturbation conditions may be hard to derive, unless the interconnection presents a specific structure [5], [12]. More importantly, since artificial time-scale separation slows down some subsystems with respect to others, it poses a fundamental limit on the convergence rate of the interconnected system.

In this work, we consider interconnected control systems in which the individual subsystems are designed and stabilized (e.g., by means of control) on separate time scales, and we are interested in preserving the overall system stability of the interconnection in a single time scale. Unlike the singular perturbation approach, we propose a single time-scale interconnection that guarantees closed-loop stability without imposing additional conditions on control parameters, nor slowing down any subsystem with respect to others. Additionally, our approach can deal with general interconnection structures, where the dynamics of each subsystem may depend on the states of all other subsystems. Our proposed interconnection can be interpreted as a transient feed-forward term in faster systems, that anticipates the dynamics of slower ones. For that, it uses the sensitivity of the fast system’s steady state with respect to the slower system’s state. Therefore, we denote this approach as sensitivity conditioning.

This new interconnection is inspired by recently proposed optimization algorithms to solve problems that are usually represented on multiple time scales: The prediction-correction algorithms for time-varying optimization [18], [19], the advance steps in nonlinear model predictive control [20], [21], and the opponent-learning awareness games [22], [23]. These algorithms use the nonlinear optimization sensitivity [24], [25] to generate feed-forward terms that improve their convergence. Our approach also relates to classic backstepping [11, Ch. 14].
in the context of overcoming time-scale separation limitations. However, unlike backstepping, our approach does not require to know a stabilizing state feedback law in closed form. Hence, our approach is implementable in cases where such a feedback law is not available.

The main contributions of this article are as follows: First, we divide the problem of designing the interconnection of two (or more) subsystems into a design problem of separate time scales and a conditioning of their interconnection. For the latter, we define the sensitivity-conditioning approach, and we show how it corresponds to additional transient signals to be exchanged between the subsystems. Second, we prove that the sensitivity-conditioning approach ensures that the interconnected single-time-scale system has the same local (and even global, under further conditions) exponential stability properties as the multiple time scales system where each subsystem evolves on a different time scale. Third, we show how some degrees of freedom in the proposed design method can be used to improve the convergence rate of the interconnected system, and we provide robustness guarantees with respect to model errors. Fourth, we demonstrate the applicability of our approach on two control-design problems: 1) cascade control [15] and 2) bilevel optimization [26]. Finally, we show how to extend our approach to multiple time scales.

The rest of this article is organized as follows. Section II presents the type of systems that we consider and motivates our sensitivity-conditioning approach. Section III introduces the sensitivity-conditioning for two interconnected systems. Sections IV and V show the application examples. Section VI shows how the sensitivity conditioning can be extended to multiple-time-scales systems, and discrete-time systems. Finally, Section VII concludes the article.

II. MOTIVATION

Consider the interconnection of two systems described by the vector fields \( f_i \) on \( x_i \in \mathbb{R}^{n_i} \) for \( i = 1, 2 \)

\[
\begin{align*}
\dot{x}_1 &= f_1(x_1, x_2) \\
\dot{x}_2 &= f_2(x_1, x_2).
\end{align*}
\]

The study (or design) of such interconnection is challenging in general. One way to tackle these analyses or design problems is to assume that the two subsystems in (1) evolve on separate time scales: \( x_2 \) evolves on a faster time \( \tau \), where \( x_1 \) is constant, while \( x_1 \) evolves on a slower time \( t \), where the dynamics of \( x_2 \) are replaced by the algebraic equation \( f_2(x_1, x_2) = 0 \). This two-time-scale interconnection is represented with a differential-algebraic-equation system \( \Sigma_1 \), and a boundary-layer system \( \Sigma_2 \)

\[
\begin{align*}
\Sigma_1 : \dot{x}_1 &= f_1(x_1, x_2) \quad \text{s.t.} \quad f_2(x_1, x_2) = 0 \quad (2a) \\
\Sigma_2 : \frac{dx_2}{d\tau} &= f_2(x_1, x_2) \quad \text{s.t.} \quad \frac{dx_1}{d\tau} = 0. \quad (2b)
\end{align*}
\]

Many interconnected systems become simpler to design and control when the two subsystems are assumed to evolve on two separate time scales as in (2). Classical examples are adaptive control [27] and cascade control systems [15], where the state of one system is used as input to the other system, i.e., \( x_2 \) to control \( x_1 \), or nested iterative numerical algorithms (e.g., in optimization). In the rest of the article, we assume that the analysis and design of the two-time-scale system (2) are tractable problems, and we provide some examples of how this is done for specific applications in Sections IV and V.

Clearly, any statements on the steady-state behavior and the stability of the two-time-scale system (2) do not automatically hold true for the original single-time-scale system (1). One standard way to ensure that the properties of (2) extend to (1) is to enforce a sufficient time-scale separation between the two subsystems and then employ the tools of singular perturbation analysis [11, Ch. 11]. Under the assumption that \( f_2(x_1, \cdot) \) has a finite number of isolated roots \( x_2^i(x_1) \), one can define the standard singular perturbation conditioned system

\[
\begin{align*}
\dot{x}_1 &= f_1(x_1, x_2) \\
\epsilon \dot{x}_2 &= f_2(x_1, x_2)
\end{align*}
\]

where \( 0 < \epsilon \ll 1 \) is a design parameter to enforce the desired level of time-scale separation. In the singular limit \( \epsilon \to 0 \), (3) becomes a degenerate system by Tikhonov’s Theorem [3] and reduces to (2). Singular perturbation analysis allows to guarantee that if both systems in (2) are asymptotically stable, then the conditioned interconnection (3) is also stable (and has the same equilibria) when \( \epsilon \) is below a certain threshold \( \epsilon [11, \text{Th. 11.3.4}] \). An example of a control design targeting a time-scale-separated closed-loop system as in (3) is cascaded control, e.g., in power electronics control systems, where time-scale separation does not exist naturally, but has to be imposed artificially [5]. This type of conditioning comes at a cost: As the second subsystem cannot be made arbitrarily fast in practice, the design choice of \( \epsilon \ll 1 \) necessarily slows down the first subsystem and, thus, limits the convergence rate and deteriorates the performance of the entire interconnection (3).

In this article, we propose an alternative conditioning of the interconnected system (1) without this drawback. For that, we define the conditioned interconnected system \( \Sigma \) as

\[
\Sigma : \begin{bmatrix}
M(x_1, x_2) & \dot{x}_1 \\
\dot{x}_2 & f_1(x_1, x_2)
\end{bmatrix}
\]

\[
\begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2
\end{bmatrix} = \begin{bmatrix}
f_1(x_1, x_2) \\
f_2(x_1, x_2)
\end{bmatrix}
\]

where \( M(\cdot) = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \cdot (\cdot) \) is a general nonsingular conditioning matrix, i.e., a generalized time constant, which is a design variable to be chosen. Notice that the singular perturbation conditioned (3) is a special case of (4), with a specific matrix \( M = \begin{bmatrix} I & 0 \\ 0 & \epsilon I \end{bmatrix} \), where \( I \) is the identity matrix. Nonetheless, a general \( M \) can represent a much larger class of interconnections, see Table I for an illustration. For example, it can represent the fully actuated interconnected control system

\[
\begin{align*}
\dot{x}_1 &= f_1(x_1, x_2) + u_1 \\
\dot{x}_2 &= f_2(x_1, x_2) + u_2
\end{align*}
\]

where the external control inputs \( u_i \) are active only the transient dynamics, i.e., \( u_i = 0 \) when \( f_i = 0 \), thus, preserving the steady states of the original system (1) and of the two-time-scale system (2). One example of this general conditioning technique (4) is backstepping in cascade systems [11, Ch. 14], which we will review in Section IV. Other examples appear in the design
of nested gradient algorithms for continuous-time optimization, which we will discuss in Section V.

Note that the singular-perturbation-conditioned system (3) uses a transient control action $u_2 = (\frac{1}{\epsilon} - 1)f_2(x_1, x_2)$ and $u_1 = 0$ to induce time-scale separation. In this article, we propose an alternative conditioning matrix $M(-)$ in (4) that, in the form (5), corresponds to a derivative-type control action $u_2 = M_2 f_2(x_1, x_2)$. Loosely, we propose that the $x_2$-dynamics are additionally driven by $u_2 \approx \frac{d}{dt} x_2^s(x_2(t))$, where $x_2^s(x_2)$ is the steady state of the boundary-layer system (2b) parameterized by $x_2$, i.e., $f_2(x_1, x_2^s(x_2)) = 0$. As a result, under the dynamics (5), we have $\frac{d}{dt} \|x_2(t) - x_2^s(x_2(t))\|^2 = 2(x_2(t) - x_2^s(x_2(t)))^T(f_2(x_1(t), x_2(t)) + u_2 - \frac{d}{dt} x_2^s(x_2(t)))$. Likewise, for (2b), we have $\frac{d}{dt} \|x_2(t) - x_2^s(x_2(t))\|^2 = 2(x_2(t) - x_2^s(x_2(t)))^T f_2(x_1(t), x_2(t)).$ In other words, if stability of the instantaneous steady state $x_2^s(x_1(t))$ of (2b) can be inferred by means of a quadratic Lyapunov function, so can the stability of the trajectory $x_2^s(x_1(t))$ of (5). In either case, the stability analysis of the coupled $(x_1, x_2)$ dynamics reduces to that of a cascade system, and no time-scale separation is required.

However, generally $x_2^s(x_1(t))$ and its derivative $\frac{d}{dt} x_2^s(x_1(t))$ are not available in closed form. In what follows, we show how to construct an implementable surrogate for $u_2 \approx \frac{d}{dt} x_2^s(x_1(t))$, analyze the system stability without requiring $x_2^s(x_1(t))$, and extend the argument to an arbitrary number of subsystems.

III. SENSITIVITY CONDITIONING FOR TWO SYSTEMS

A. Steady States and Sensitivity

For this section we make the following standard simplifying assumption [11, Ch. 11], which we will partially relax later in Section VI:

Assumption I: The vector fields $f_i(\cdot)$ are continuously differentiable. For every $x_1$, $f_2(x_1, \cdot) = 0$ has a single root $x_2^s$, where the partial derivative $\nabla_{x_2} f_2(x_1, x_2^s)$ is invertible.

Under Assumption I, the implicit-function theorem [28] guarantees the local existence of a continuously differentiable steady-state map $x_2^s(x_1)$, and gives the sensitivity of this steady state $x_2^s(x_1)$ with respect to $x_1$ as

$$\nabla_{x_1} x_2^s(x_1) = -\nabla_{x_2} f_2(x_1, x_2^s(x_1))^{-1} \nabla_{x_1} f_2(x_1, x_2^s(x_1)).$$

where $\nabla_{x_1} x_2^s(x_1) \in \mathbb{R}^{n_2 \times n_1}$. Even though this sensitivity is defined only at points where $x_2 = x_2^s(x_1)$, given Assumption I, its analytic expression is well defined at any point $x_2$ in a neighborhood of $x_2^s(x_1)$. This allows to define an extended sensitivity

$$S_{x_1}^2 (x_1, x_2) := -\nabla_{x_2} f_2(x_1, x_2)^{-1} \nabla_{x_1} f_2(x_1, x_2)$$

which satisfies the restriction $S_{x_1}^2 (x_1, x_2^s(x_1)) = \nabla_{x_1} x_2^s(x_1)$.

The steady-state map $x_2^s(x_1)$ allows us to redefine the differential-algebraic-equation system (2a) as a reduced-order system with reduced vector field $f_1^r(\cdot)$, so that the two-time-scale system (2) becomes

$$\begin{align}
\Sigma_1 & : \dot{x}_1 = f_1^r(x_1) := f_1(x_1, x_2^s(x_1)) \quad (7a) \\
\Sigma_2 & : \frac{dx_2}{dt} = f_2(x_1, x_2) \quad \text{s.t.} \quad \frac{dx_1}{dt} = 0. \quad (7b)
\end{align}$$

Then, each steady state $x_1^s$ satisfying $f_1^r(x_1^s) = 0$, defines a steady state $(x_1^s, x_2^s(x_1^s))$ for (7), and by [11, Corollary 4.3] it is a locally exponentially stable steady state of the two-time-scale system (7) if and only if $\nabla_{x_2} f_2(x_1^s, x_2^s(x_1^s))$ and

$$\nabla_{x_1} f_1^r(x_1^s) = \nabla_{x_1} f_1(x_1^s, x_2^s(x_1^s)) + \nabla_{x_2} f_1^r(x_1^s, x_2^s(x_1^s))$$

have eigenvalues with negative real part, and unstable if any of these matrices has any eigenvalue with positive real part.

Remark I: Both the singular perturbed (3) and the two-time-scale systems (7) have the same steady state $(x_1^s, x_2^s(x_1^s))$, but their local exponential stability properties may differ given the value of $\epsilon$, because their Jacobians and, thus, their local linearizations, may have different eigenvalues, see Table I. For example, consider the linear system $f_1(x_1, x_2) = x_1 - 2x_2, f_2(x_1, x_2) = \frac{1}{\epsilon} x_1 - \frac{1}{\epsilon} x_2$, thus, $x_2^s(x_1) = 1, \nabla_{x_1} x_2^s(x_1) = 1, f_1^r(x_1^s) = -x_1^s, and (x_1^s, x_2^s(x_1^s)) = (0, 0).$ Then, (7) is exponentially stable, since $J_2 = \nabla_{x_2} f_2(0, 0) = -\frac{1}{\epsilon} < 0$ and $J_1 = \nabla_{x_1} f_1^r(0, 0) = -1 < 0.$ The Jacobian of (3) in Table I has indeed negative eigenvalues for $\epsilon < \frac{1}{2},$ but positive ones for $\epsilon > \frac{1}{2}.$ Thus, (3) is exponentially stable if and only if $\epsilon < \frac{1}{2}.$

B. Sensitivity-Conditioning Interconnection

Here, we present an alternative interconnection in (4), that can preserve the steady state $(x_1^s, x_2^s(x_1^s))$ of the two-time-scale system (7) and its stability, without the need of a sufficiently large time-scale separation via a singular parameter $\epsilon$. This interconnection uses a sensitivity-conditioning matrix $M = \begin{bmatrix} I & 0 \\ -S_{x_1}^2 & I \end{bmatrix}$, graphically presented in Table I

$$\begin{bmatrix} I & 0 \\ -S_{x_1}^2 & I \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} f_1(x_1, x_2) \\ f_2(x_1, x_2) \end{bmatrix}. \quad (9)$$

Instead of accelerating the second subsystem as in (3), this conditioning matrix $M$ contains an off-diagonal term that changes the dynamics of the second subsystem to $\dot{x}_2 = f_2(x_1, x_2) + S_{x_1}^2(x_1, x_2^s(x_1)) f_1(x_1, x_2),$ i.e., using the control input $u_2 = S_{x_1}^2(x_1, x_2^s(x_1)) f_1(x_1, x_2)$ in (5). Intuitively, there are now two components in the vector field of $\dot{x}_2$: $f_2(x_1, x_2)$ drives $x_2$ to the steady state $x_2^s(x_1)$, while the sensitivity conditioning $S_{x_1}^2(x_1, x_2^s(x_1)) f_1(x_1, x_2)$ can be interpreted as a feed-forward term.
TABLE I
COMPARISON OF INTERCONNECTIONS $M$

| Case | Two time scales (7) | Sing. perturbed (3) | Sensitivity cond. (9) | Generalized S-C (16) |
|------|----------------------|---------------------|----------------------|----------------------|
| System $\Sigma$ | $\dot{x}_1 = f_1(x_1)$ | $\dot{x}_2 = f_2(x_1, x_2)$ | $M \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} f_1(x_1, x_2) \\ f_2(x_1, x_2) \end{bmatrix}$ |
| Matrix $M$ | $\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$ | $\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$ | $\begin{bmatrix} I & 0 \\ -S_{x_1} x_2 \end{bmatrix}$ | $\begin{bmatrix} H_{x_1}^{-1} & 0 \\ 0 & H_{x_2}^{-1} \end{bmatrix}$ |
| Block diagram of $M$ in Fig. 1 | | | | |
| Jacobian $J$ at $(x_1^2(x_1^0), x_2^2(x_1^0))$ | $J_1 = \nabla x_1 f_1^t$ | $\begin{bmatrix} \nabla x_1 f_1 & \nabla x_2 f_1 \\ \frac{1}{x_1} \nabla x_1 f_2 & \frac{1}{x_2} \nabla x_2 f_2 \end{bmatrix}$ | $\nabla x_2 f_1$ | $\begin{bmatrix} H_1 \nabla x_1 f_1^t H_1 \nabla x_2 f_1 \\ 0 \\ H_2 \nabla x_2 f_2 \end{bmatrix}$ |
| Eigenvalues $\lambda$ & local stability | [11, Corollary 4.3]: Exp. stable if and only if $\lambda$ of Jacobians $J_i$ have negative real part. | Remark 1: Different $\lambda$ can be unstable even if (7) stable. Stable if $\epsilon < \epsilon_0$. | Proposition 2 & Cor. 1: Similar to block-diagonal $J$, same $\lambda$ as (7), thus, same local stability. | Proposition 4 & Cor. 3: Preserving stability, $\lambda$ of $J$ can have lower negative real part than (7) and (9), thus, faster convergence. |

anticipating the change of $x_2^2(x_1)$ due to the dynamics $\dot{x}_1 \neq 0$. This second term affects the transient behavior only and vanishes at steady state.

Given Assumption 1, local existence and uniqueness [11, Th. 3.1] of a solution $x(t)$ for (9) are guaranteed if:

Assumption 2: The vector field $f_2(x_1, x_2) + S_{x_1}^2(x_1, x_2) f_1(x_1, x_2)$ is locally Lipschitz continuous.

For more insight on the benefits of (9), we advance some results, that specialize the more general Theorem 1 (presented later in Section VI) to the case of two interconnected systems. The first proposition shows that the singleton $\{x_2^2(x_1)\}$ is a positively invariant set, i.e., once $x_2$ hits the steady state $x_2^2(x_1)$, it remains at $x_2^2(x_1)$ even if $\dot{x}_1 \neq 0$.

Proposition 1 (Positive invariance): Consider the dynamics of $x_2$ in (9) initialized at time $t_0$

$$\dot{x}_2 = f_2(x_1, x_2) + S_{x_1}^2(x_1, x_2) \dot{x}_1 s.t. x_2(t_0) = x_2^0(x_1(t_0)).$$

Then, $x_2(t) = x_2^0(x_1(t))$ is the unique solution on the open domain of existence.

Proof: First, note that $x_2(t) = x_2^0(x_1(t))$ satisfies (10)

$$\dot{x}_2(t) = f_2(x_1(t), x_2(t)) + S_{x_1}^2(x_1(t), x_2(t)) \dot{x}_1(t) = f_2(x_1(t), x_2^0(x_1(t)))$$

$$+ S_{x_1}^2(x_1^0(x_1^0(t)), x_2^0(x_1^0(t))) \dot{x}_1(t).$$

$$\frac{dx_2^0(x_1(t))}{dt} = \frac{\partial \nabla x_1, x_2^0}{\nabla x_2 f_2}.$$ Local existence and uniqueness of a solution is guaranteed by Assumption 2. Thus, $x_2(t) = x_2^0(x_1(t))$ is the unique solution on this domain of existence, because the derivatives and initial conditions of $x_2(t)$ and $x_2^0(x_1(t))$ coincide for $t \geq t_0$. Moreover, the sensitivity conditioning (9) allows to preserve the local stability of the two-time-scale system (7):

Proposition 2 (Local stability): At a steady state $(x_1^*, x_2^*(x_1^*))$, the Jacobian $J$ of (9) satisfies

$$J \approx \begin{bmatrix} \nabla x_1 f_1^t(x_1^*) & \nabla x_2 f_1(x_1^*, x_2^*(x_1^*)) \\ 0 & \nabla x_2 f_2(x_1^*, x_2^*(x_1^*)) \end{bmatrix}$$

(11)

where $f_1^t$ is the reduced vector field from (7a), and $\approx$ denotes similarity, i.e., related by a similarity transformation.

Proof: To calculate the Jacobian $J$ of (9) at $(x_1^*, x_2^*(x_1^*))$, we invert $M$ as $M^{-1} = \left[ \begin{bmatrix} I \\ S_{x_1}^2 \end{bmatrix} \right]$, take derivatives, and evaluate them at steady state, so that $f_i(x_1^*, x_2^*(x_1^*)) = 0$.

$$J = M^{-1} \begin{bmatrix} \nabla x_1 f_1 & \nabla x_2 f_1 \\ \nabla x_1 f_2 & \nabla x_2 f_2 \end{bmatrix}.$$
where for clarity we omit the evaluation point \((x_1^2, x_2^2(x_1^1))\) in the notation. This \(J\) is similar to \(J := MJM^{-1}\), where
\[
J \sim \tilde{J} = MJM^{-T} = \begin{bmatrix}
\nabla_{x_1} f_1 & \nabla_{x_2} f_1 \\
\nabla_{x_1} f_2 & \nabla_{x_2} f_2
\end{bmatrix}
\]
and
\[
M^{-1} = \begin{bmatrix}
\nabla_{x_1} f_1 & \nabla_{x_2} f_1 \\
0 & \nabla_{x_2} f_2
\end{bmatrix}
\]

**Corollary 1:** The Jacobians of the two-time-scale system (7) and the sensitivity-conditioning conditioned system (9) have the same eigenvalues and, thus, the same local stability properties.

**Remark 2:** The cancellation of one off-diagonal term in (11) is due to the sensitivity definition in (6), and will also play a crucial role in the proofs of the results to come. Essentially, the role of the sensitivity conditioning is to turn a closed-loop into a cascade system from the viewpoint of stability analysis, see also the later Remark 4.

Propositions 1 and 2 and Corollary 1 establish that invariance and local exponential stability of \(x_2^2(x_1)\) are preserved from the two-time-scale system (7) in the single-time-scale sensitivity-conditioning one (9). Furthermore, these results can be extended to contraction regions satisfying the following:

**Assumption 3 (Contraction region [29]):** There exists \(\eta_2 > 0\) and an open ball \(B_{\eta_2}(x_2^2(x_1)) = \{x_2 | \|x_2 - x_2^2(x_1)\| < \eta_2\}\) centered at \(x_2^2(x_1)\), with a positive radius \(\eta_2 > 0\), and a metric defined by a constant-symmetric-positive definite \(P > 0\), such that for \(\eta_2 > 0\), \(P > 0\), such that for \(x_2 \in B_{\eta_2}(x_2^2(x_1))\) it holds uniformly for all \(x_1\), that
\[
P_2 \nabla_{x_2} f_2(x_1, x_2) + \nabla_{x_2} f_2(x_1, x_2)^T P_2 \leq -\eta_2 P_2.
\]

Under this Assumption 3, \(B_{\eta_2}(x_2^2(x_1))\) is a contraction region for the boundary-layer system (2b) for all \(x_1\), within which the invariant set \(x_2^2(x_1)\) (see Proposition 1) is exponentially stable [29, Th. 2]. Then, such a contraction region is also preserved under the sensitivity conditioning (9):

**Proposition 3 (Stability with a contracting boundary layer):** Under Assumption 3 the following holds:

1. The sensitivity-conditioning interconnection (9) is well defined for \(x_2 \in B_{\eta_2}(x_2^2(x_1))\), i.e., \(\nabla_{x_2} f_2(x_1, x_2)\) is uniformly invertible in \(B_{\eta_2}(x_2^2(x_1))\).

2. Furthermore, there exists \(\eta_2 > 0\), such that, if \(x_2(0) \in B_{\eta_2}(x_2^2(x_1(0)))\), then \(x_2(t) \in B_{\eta_2}(x_2^2(x_1(t)))\) for all \(t > 0\), and \(x_2^2(x_1(t))\) is a locally exponentially stable trajectory for \(x_2\) in (9), i.e., there exists \(\delta > 0\), \(K > 0\) such that
\[
\|x_2(t) - x_2^2(x_1(t))\| \leq K \|x_2(0) - x_2^2(x_1(0))\| e^{-\delta t}.
\]

2. Additionally, assume that \(x_1^2\) is an asymptotically-stable steady state of the reduced-order system (7a), that the ball \(B_{\eta_2}(x_1^2)\) is in its region of attraction, and that \(f_1^2(\cdot)\) is continuously differentiable in the closure of \(B_{\eta_2}(x_1^2)\).

Then, there exists \(\eta_1 > 0\), \(\eta_2 > 0\), such that, if \(x_1(0) \in B_{\eta_1}(x_1^2)\) and \(x_2(0) \in B_{\eta_2}(x_2^2(x_1(0)))\), then \(x_1(t) \in B_{\eta_1}(x_1^2)\), and \(x_1^2\) is asymptotically stable under the sensitivity-conditioning interconnection (9).

**Proof:** We use the following technical result:

**Lemma 1:** Consider a system \(\dot{x} = f(x)\), with steady-state \(x^s\) and a continuous differentiable \(f(\cdot)\). If there exists a radius \(r > 0\), a symmetric positive definite matrix \(P > 0\), and a parameter \(\eta > 0\), such that \(P\nabla_x f(x) + (\nabla_x f(x))^T P^T \leq -\eta P\) for \(x \in B_r(x^s)\), then in \(B_r(x^s)\) it holds that:

1. The inverse of the Jacobian \(\nabla_x f(x)\) exists and is bounded:
\[
\|\nabla_x f(x)^{-1}\| \leq \frac{2\max(\|P\|)}{\delta_{\text{min}}(\|P\|)}.
\]

2. The vector field \(f(x)\) is lower bounded:
\[
\|f(x)\|_P \geq \frac{\delta_{\text{max}}(\|P\|)}{\|x - x^s\|_P}
\]

Here \(\delta_{\text{min}}(\cdot)\) and \(\delta_{\text{max}}(\cdot)\) denote the minimum and maximum eigenvalues, respectively.

**Proof:** See Appendix A. We remark that the second result can be seen as a particular case of [30, Prop. 3].

1. First, note that Lemma 1 assures nonsingularity of \(\nabla_x f(x)\). Consider the following Krasovskii–Lyapunov function \(V_2(x_1, x_2) = \|f_2(x_1, x_2)\|_P^2\) [31, Ch. 5] for \(x_2\) in (9). Since \(P_2 \nabla_{x_2} f_2 + \nabla_{x_2} f_2^T P_2 \leq -\eta_2 P_2\), under the sensitivity-conditioning dynamics (9) we have
\[
\dot{V}_2 = f_2^T P_2 \nabla_{x_1} f_2 \dot{x}_1 + f_2^T P_2 \nabla_{x_2} f_2 \dot{x}_2
\]
\[
= (f_2^T P_2 \nabla_{x_1} f_2 x_1 + f_2^T P_2 \nabla_{x_2} f_2 x_2)^T
\]
\[
\leq \eta_2^2 P_2 \nabla_{x_2} f_2 + \nabla_{x_2} f_2^T P_2 \nabla_{x_2} f_2 \leq -\eta_2 V_2.
\]

Hence, \(\|f_2(x_1(t), x_2(t))\|_P^2 \leq \|f_2(x_1(0), x_2(0))\|_P^2 e^{-\eta_2 t}\). Since \(f_2(\cdot)\) is continuously differentiable, it is locally Lipschitz continuous in \(B_{\eta_2}(x_2^2(x_1))\) with some constant \(L_{f_2}\), and by invoking Lemma 1 we have
\[
\|x_2(t) - x_2^2(x_1(t))\|_P^2 \leq \frac{4}{\eta_2^2 \delta_{\text{min}}(P_2)} \|f_2(x_2(t), x_1(t))\|_P^2
\]
\[
= \eta_2^2 \delta_{\text{max}}(P_2) e^{-\eta_2 t} \|x_2(0) - x_2^2(x_1(0))\|_P^2
\]

where \(r_0 \leq \eta_2 \sqrt{\frac{\delta_{\text{max}}(P_2)}{\delta_{\text{min}}(P_2)}} \leq r_2\). Hence, if \(\|x_2(0) - x_2^2(x_1(0))\| < r_2\), then \(x_2 \in B_{\eta_2}(x_2^2(x_1(t)))\) for all \(t > 0\), and \(x_2^2\) converges exponentially to \(x_2^2(x_1)\) under (9), despite the varying \(x_1\).
Consider any \( \delta \in (0, r_1) \), and define \( r_{1,0} := \alpha_2^{-1}(\alpha_1(r_1 - \delta)) < r_1 \) and

\[
\bar{r}_{2,0} := \min \left( r_{2,0}, \frac{\alpha_3(r_{1,0})}{\alpha_4(r_1 - \delta)r_{2,1}/r_{2,0}} \right).
\]

If \( x_1(0) \in B_{r_{1,0}}(x_1^*) \), then \( V_1(x_1(0)) \leq \alpha_1(r_1 - \delta) \); and if \( x_2(0) \in B_{r_{2,0}}(x_2^*(x_1(0))) \), then \( V_1 \leq 0 \) whenever \( V_1 = \alpha_1(r_1 - \delta) \). Hence, \( V_1(x_1(t)) < \alpha_1(r_1) \) and \( \|x_1(t) - x_1^*\|^2 < r_1^2 \) for all \( t > 0 \). Furthermore, for any \( \epsilon \in (0, 1) \), it holds that

\[
V_1 \leq -\alpha_3(\|x_1 - x_1^*\|^2) + L_{f_1} \alpha_4(\|x_1 - x_1^*\|^2)\|x_2 - x_2^*(x_1)\|^2 \\
\leq -\alpha_3(\|x_1 - x_1^*\|^2) - (1 - \epsilon)\alpha_3(\|x_1 - x_1^*\|^2) \\
+ L_{f_1} \alpha_4(\|x_1 - x_1^*\|^2) r_{2,0}^2 e^{-\epsilon_2 t}\|x_2(0) - x_2^*(x_1(0))\|^2 \\
\leq -\alpha_3(\|x_1 - x_1^*\|^2)
\]

where the last inequality holds while \( \|x_1 - x_1^*\|^2 \geq \alpha_3^{-1}(\frac{1}{L_{f_1} \alpha_4(\|x_1 - x_1^*\|^2)} e^{-\epsilon_2 t}\|x_2(0) - x_2^*(x_1(0))\|^2) \). Hence, \( x_1^* \) is asymptotically stable, because it is input-to-state stable [11, Th. 4.18] with respect to a vanishing input, see also [11, Lemma 4.7].

Remark 3 (Connection to contraction theory): The exponential stability of the boundary-layer system (2b) for a constant \( x_1 \) has a standard assumption in the context of singular-perturbation analysis [11, Th. 11.4 and after], and exponential stability implies the existence of a contraction region [29, Reverse Th. 2]. Proposition 3 establishes that the boundary-layer exponential stability can be preserved in the single-time-scale interconnection (4) using the sensitivity conditioning (9), independently of \( x_1 \). Essentially, the sensitivity-conditioning (9) turns a system \( \Sigma_2 \) that is only contracting under a constant \( x_1 \), as in (2b), into a partially-contracting system (9) in \( x_2 \) [32, Def. 1] under a time-varying \( x_1 \). Then, \( \|x_2 - x_2^*(x_1)\|^2 \) becomes an exponentially-decaying perturbation for \( x_1 \) in (9) and, thus, asymptotic stability of the reduced-order system (7a) can be preserved in (9) under some additional conditions.

Remark 4 (Connection to backstepping): The role of the sensitivity term \( S_{x_2} \) in (9) is to cancel a cross term in the stability analysis of \( x_2 \) that appears under a time-varying \( x_1 \), see (12) and the proofs of Propositions 2 and 3. In other words, the sensitivity conditioning (9) is turning an interconnected system (4) into a cascaded one from the viewpoint of stability analysis, see Remark 2. In a more general setting for Proposition 3, we could assume that the boundary-layer system (2b) is asymptotically stable with a general Lyapunov function \( V_2(x_1, x_2) \) that is positive definite with respect to \( \|x_2 - x_2^*(x_1)\| \), and satisfies \( \nabla_{x_2} V_2^T f_2 \leq 0 \) uniformly over some additional conditions.

Then, under the sensitivity-conditioning dynamics (9) we would have \( V_2 \leq \nabla_{x_2} V_2^T f_2 + (\nabla_{x_2} V_2^T S_{x_2} - \nabla_{x_2} V_2^T) x_1 \leq (\nabla_{x_2} V_2^T S_{x_2} - \nabla_{x_2} V_2^T) x_1 \), which can be cancelled by choosing an appropriate sensitivity \( S_{x_2} \). In Proposition 3, we considered \( V_2 = \|f_2\|^2 \). An other option would be \( V_2 = \|x_2 - x_2^*(x_1)\|^2 \), which requires the alternative sensitivity \( \nabla_{x_2} V_2^T (x_1) = \nabla_{x_2} V_2^T (x_1) - \nabla_{x_2} V_2^T (x_1) x_1 \), to cancel the term \( \nabla_{x_2} V_2^T S_{x_2} - \nabla_{x_2} V_2^T = 2 \|x_2 - x_2^*(x_1)\|^2 - \nabla_{x_2} V_2^T (x_1) x_1 \). This sensitivity \( \nabla_{x_2} V_2^T (x_1) \) can be interpreted as a backstepping-like approach [11, Ch. 14] to cancel the dynamics of \( x_1 \) not in \( V_2 \), and again turn (4) in a cascaded system from the viewpoint of stability analysis. Note, that for this sensitivity, similar results as in Propositions 2 and 3 can be derived under suitable assumptions. However, implementing the corresponding interconnection \( M = -\nabla_{x_2} x_2^*(x_1) \) [or equivalently \( u_2 = \nabla_{x_2} x_2^*(x_1) f_1(x_1, x_2) = \frac{dt}{dt} x_2^*(x_1(t)) \)] may not be feasible, since it requires a closed-form expression for \( x_2^*(x_1) \) to evaluate \( \nabla_{x_2} x_2^*(x_1) \). Such a closed-form expression for \( x_2^*(x_1) \) may be available in special cases, see cascade control in Section IV, but not in general, see bilevel optimization in Section V. On the other hand, choosing \( V_2 = \|f_2\|^2 \) results in the sensitivity in (6), which does not require to know \( x_2^*(x_1) \). In this context, the sensitivity conditioning (9) acts as an implementable substitute for such a backstepping-like approach, with the same local properties.

Corollary 2 (Global-exponential stability): Assume that the vector fields \( f_i(x_1, x_2) \) are Lipschitz continuous, and there exist \( P_i > 0 \) and \( \eta_i > 0 \) such that the following contraction conditions hold globally for all \( x_1 \) and \( x_2 \):

\[
P_1 \nabla_{x_1} f_i^T (x_1) + (\nabla_{x_1} f_i^T (x_1))^T P_1 \leq -\eta_i P_1 \]

\[
P_2 \nabla_{x_2} f_2^T (x_1, x_2) + \nabla_{x_2} f_2^T (x_1, x_2)^T P_2 \leq -\eta_2 P_2.
\]

Then, \( (x_1^*, x_2^*(x_1^*)) \) is a globally exponentially stable steady state of both the two-time-scale system (7) and the sensitivity-conditioning interconnection (9).

Proof: If the contraction conditions (15) hold, (7) is globally exponentially stable [29, Th. 2]. For (9), consider now the Lyapunov function \( V(x_1, x_2) = V_1(x_1) + \theta V_2(x_1, x_2) \), where \( \theta > 0 \). \( \nabla_{x_1} V_1(x_1) = \|f_1(x_1)\|^2 \) and \( \nabla_{x_1} V_2(x_1, x_2) = \|f_2(x_1, x_2)\|^2 \).

From (12) we have \( \dot{V}_2 \leq -\eta_2 V_2 \) then

\[
\dot{V}_1 = (f_1^T P_1 \nabla_{x_1} f_1^T + \|f_1\|^2 P_1 \nabla_{x_1} f_1^T)^T \\
(9) \leq (f_1^T P_1 \nabla_{x_1} f_1^T + \|f_1\|^2 P_1 \nabla_{x_1} f_1^T f_1 - f_1 f_1^T)^T \\
+ 2(f_1^T P_1 \nabla_{x_1} f_1^T f_1 - f_1 f_1^T)^T \\
\leq -\eta_1 \|f_1\|^2 P_1 + 2L_{f_1} \|f_1\|^2 P_1 \|f_1 - f_1\|^2 P_1 \\
\leq -\eta_1 \|f_1\|^2 P_1 + 2L_{f_1} \sqrt{\frac{\max_{i}(\eta_i)}{\min_{i}(\eta_i)} P_1 \|f_1\|^2 P_1} \|f_1\|^2 P_1 \|x_2 - x_2^*(x_1)\|^2 P_2 \\
\leq -\eta_1 \|f_1\|^2 P_1 + \sqrt{\frac{\max_{i}(\eta_i)}{\min_{i}(\eta_i)} P_1 \|f_1\|^2 P_1} \|f_1\|^2 P_1 \|x_2 - x_2^*(x_1)\|^2 P_2 \leq -\eta_1 \|f_1\|^2 P_1 + \sqrt{\frac{\max_{i}(\eta_i)}{\min_{i}(\eta_i)} P_1 \|f_1\|^2 P_1} \|f_1\|^2 P_1 \|x_2 - x_2^*(x_1)\|^2 P_2.
\]
Hence, \( V \leq \frac{1}{2} \| f_h(\eta_2) \| \) for some \( \zeta > 0 \), by choosing \( \theta > \frac{2}{\eta_1/\eta_2} \).

### C. Accelerated Sensitivity Conditioning

The design of the conditioning matrix \( M(x_1, x_2) \) in (4) offers to generalize the sensitivity conditioning (9) to introduce additional degrees of freedom and achieve a better performance of the interconnection, e.g., a faster convergence. Consider two uniformly positive definite matrices \( H_1(x_1, x_2) \) and \( H_2(x_1, x_2) \) and, a generalized sensitivity conditioning

\[
H(x_1, x_2)^{-1} \begin{bmatrix} \mathbf{I} & 0 \\ -S_x^2(x_1, x_2) & \mathbf{I} \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} f_1(x_1, x_2) \\ f_2(x_1, x_2) \end{bmatrix} \tag{16}
\]

where \( H(x_1, x_2) = \begin{bmatrix} H_1(x_1, x_2) & 0 \\ 0 & H_2(x_1, x_2) \end{bmatrix} \).

**Proposition 4 (Extension of Propositions 1, 2, and Corollary 2):** The generalized sensitivity conditioning (16) satisfies.

1. The singleton \( \{x_2(x_1)\} \) is a positively-invariant set under the sensitivity-conditioning dynamics (16).
2. At steady state \( (x_1^s, x_2^s (x_1^s)) \), the Jacobian of (16) satisfies

\[
J \approx \begin{bmatrix} H_1 \nabla x_1 f_1 & H_1 \nabla x_2 f_1 \\ 0 & H_2 \nabla x_2 f_2 \end{bmatrix}.
\]

3. If the vector fields \( f_i(x_1, x_2) \) are Lipschitz continuous, and there exists \( P_i > 0 \) and \( \eta_i > 0 \) such that

\[
P_1 H_1 \nabla x_1 f_1 + \langle \nabla x_1 f_1 \rangle^T H_1^T P_1 \leq -\eta_1 P_1 \]

\[
P_2 H_2 \nabla x_2 f_2 + \langle \nabla x_2 f_2 \rangle^T H_2^T P_2 \leq -\eta_2 P_2
\]

then \( (x_1^s, x_2^s (x_1^s)) \) is a globally exponentially stable steady state of the generalized sensitivity-conditioning (16).

**Proof:** The proof follows analogous steps as the ones for Propositions 1, 2, and Corollary 2. \( \blacksquare \)

**Corollary 3 (Accelerated sensitivity conditioning):** If the two-time-scale system (7) is locally exponentially stable, i.e., \( J_1 = \nabla x_1 f_1 (x_1) \) and \( J_2 = \nabla x_2 f_2 (x_1, x_2^s (x_1)) \) have eigenvalues with strictly negative real part, the generalized sensitivity-conditioning system (16) is locally exponentially stable if using positive scalars \( h_1 > 0 \) and \( H_1(x_1, x_2) = h_1 I \). Moreover, the exponential convergence rate is improved for \( h_1 > 1 \).

See Table I for a comparison of the sensitivity-conditioning approach (9) and (16), the two-time-scale system (7), and the singular perturbed (3), summarizing these results.

This way, the need of artificially slowing down one subsystem (and, consequently, their interconnection) through a singular perturbation (3) is removed. However, a disadvantage of the sensitivity conditioning is that it could produce large inputs \( u \) for \( \Sigma_2 \) in (5), even larger if using the generalization in (16), which changes the sensitivity-conditioning term to \( S_x^2(x_1, x_2) H_1(x_1, x_2) f_1(x_1, x_2) \). This sensitivity-conditioning term could even become unrealizable in systems with control saturation in \( u \). On the other hand, if the two-time-scale system (7) is locally stable, the generalization (16) can be chosen as \( H_1(x_1, x_2) = \epsilon \) with a sufficiently small \( \epsilon \), see Corollary 3. Then, \( \Sigma_1 \) can be slowed down as with a singular-perturbation term, and the sensitivity-conditioning term can be made realizable. This interpretation suggests that singular perturbation (3) and sensitivity conditioning (9) are not mutually exclusive, but can be combined. Interestingly, Corollary 3 also allows to choose arbitrary time scales, e.g., \( \Sigma_1 \) faster than \( \Sigma_2 \) by choosing \( h_1 \gg h_2 \), and still preserve the stability of (7).

### D. Robust Sensitivity Conditioning

The sensitivity-conditioning system (9) requires a precise knowledge of the vector fields, essentially the model of the system, to evaluate the matrix \( S_x^2(x_1, x_2) \). Here, we analyze the implications of model errors: Assume that instead of \( S_x^2(x_1, x_2) \), only an approximation \( \hat{S}_x^2(x_1, x_2) \) is available, and consider the approximated sensitivity-conditioning

\[
\begin{bmatrix} \mathbf{I} & 0 \\ -\hat{S}_x^2(x_1, x_2) & \mathbf{I} \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} f_1(x_1, x_2) \\ f_2(x_1, x_2) \end{bmatrix} \tag{17}
\]

which also preserves the steady state \( (x_1^s, x_2^s (x_1^s)) \).

Under Assumptions 1 and 2, we can derive a strong robustness certificate in the form of input-to-state stability [11, Def. 4.7]:

**Proposition 5 (Input-to-state stability):** Consider the same conditions as in Corollary 2, and define the error

\[
\xi := (S_x^2(x_1, x_2) - \hat{S}_x^2(x_1, x_2)) f_1(x_1, x_2).
\]

Then, the approximated sensitivity-conditioning system (17) is input-to-state stable with respect to \( \xi \).

**Proof:** Since \( (x_1, x_2^s(x_1)) \) under (9) is exponentially stable under the conditions of Corollary 2, it is input-to-state stable in (17) with respect to \( \xi \) [11, Lemma 4.6]. \( \blacksquare \)

If the contraction conditions (15) in Corollary 2 do not hold globally, a local result along the line of Proposition 3 can be derived based on the local-exponential stability of \( x_2^s(x_1) \).

### IV. Example I: Cascade Control

Consider a standard cascade control architecture [15], see Fig. 2, with a fast-inner closed-loop system \( \Sigma_2 \), and a slow-outter closed-loop system \( \Sigma_1 \), both with plants \( P_i \) and controllers \( C_i \). More concretely, consider an example with two linear scalar first-order systems

\[
P_1 : \dot{x}_1 = a_1 x_1 + b_1 u_1, \quad u_1 = x_2
\]

\[
P_2 : \dot{x}_2 = a_2 x_2 + b_2 u_2
\]

where all parameters are real valued. Proportional Integral (PI) controllers are typically used for \( C_1, C_2 \)

\[
C_1 : x_2' = u_1' = -\frac{1}{b_1} (a_1 x_1 + K_{p,1} (x_1 - x_1^s) + K_{l,1} \zeta_1)
\]

\[
\zeta_1 = (x_1 - x_1^s)
\]

\[
C_2 : u_2 = -\frac{1}{b_2} (a_2 x_2 + K_{p,2} (x_2 - x_2^s) + K_{l,2} \zeta_2)
\]

\[
\zeta_2 = (x_2 - x_2^s)
\]

where \( \zeta \) are the integral error states, \( K_{p,i}, K_{l,i} \) are control gains to be determined, and the terms \( a_i, x_i \) are feed-forward terms to
Fig. 2. Block diagram of a cascade control.

Fig. 3. Block diagram of cascade PI control (19), with plants $P_i$ and controllers $C_i$ in Laplace domain. The additional sensitivity-conditioning elements in (20) are depicted in orange blocks and dotted thicker arrows.

cancel the system dynamics. If the systems $\Sigma_i$ had a time-scale separation as (7), the resulting interconnected system, with states $x_i, \zeta_i$ for each $\Sigma_i$, can be expressed as

\[
\begin{align*}
\Sigma_1: \dot{x}_1 &= -K_{P,1}(x_1 - x_1^s) - K_{I,1}\zeta_1 \\
\dot{\zeta}_1 &= (x_1 - x^s_1) \\
\Sigma_2: \frac{dx_2}{d\tau} &= -K_{P,2}(x_2 - x^s_2) - K_{I,2}\zeta_2 \\
\frac{d\zeta_2}{d\tau} &= (x_2 - x^s_2) \\
x^s_2 &= \frac{1}{b_2} (a_1x_1 + K_{P,1}(x_1 - x_1^s) + K_{I,1}\zeta_1) \quad (19)
\end{align*}
\]

which admits the globally asymptotically stable steady state $x^s_1 = x^*_1, x^s_2 = x^*_2 = u^*_1 = \frac{-a_1x^*_1}{b_1}, \zeta^*_1 = 0, \zeta^*_2 = 0$ for positive gains $K_{P,i} > 0, K_{I,i} > 0$. See Fig. 3 for a block-diagram representation of this control architecture.

Remark 5: The feed-forward control inputs in (18) can also be implemented using the references $x^*_1, x^*_2 = u^*_1$ instead of the states $x_1, x_2$, to compensate the plant dynamics. Then, the conditions for asymptotic stability of (19) are $K_{P,i} > a_i$ and $K_{I,i} > 0$. These controllers (18) may also not include any feed-forward compensation at all. Then, the conditions for asymptotic stability of (19) are $a_i - b_iK_{P,i} < 0, K_{I,i} > 0$. In either case, all our subsequent results hold with minor adjustments.

To preserve the stability of the time-scale-separated cascaded system with controllers (19) in a single time scale we apply the sensitivity conditioning (9)

\[
\begin{align*}
\dot{x}_1 &= a_1x_1 + b_1x_2 \\
\dot{\zeta}_1 &= (x_1 - x^*_1) \\
x^*_2 &= u^*_1 = \frac{1}{b_1} (a_1x_1 + K_{P,1}(x_1 - x^*_1) + K_{I,1}\zeta_1) \\
\dot{x}_2 &= -K_{P,2}(x_2 - x^*_2) - K_{I,2}\zeta_2 + S^{\mathbf{g}}_{[x_1, \zeta_1]} \begin{bmatrix} \dot{x}_1 \\ \dot{\zeta}_1 \end{bmatrix} \\
\dot{\zeta}_2 &= (x_2 - x^*_2) + S^{\mathbf{g}}_{[x_1, \zeta_1]} \begin{bmatrix} \dot{x}_1 \\ \dot{\zeta}_1 \end{bmatrix} \quad (20)
\end{align*}
\]

where

\[
\begin{bmatrix} S^{\mathbf{g}}_{[x_2, \zeta_2]} \\ S^{\mathbf{g}}_{[x_1, \zeta_1]} \end{bmatrix} = -\begin{bmatrix} -K_{P,2} & -K_{I,2} \\ 1 & 0 \end{bmatrix}^{-1} \begin{bmatrix} -K_{P,2} & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} -a_1 + K_{P,1} & b_1 \\ b_1 & 0 \end{bmatrix}
\]

Note, that $S^{\mathbf{g}}_{[x_1, \zeta_1]} = [0 \, 0]$ is due to $\zeta_2^* = 0$ for all $x_1, \zeta_1$, for a stable inner system $\Sigma_2$. In summary, the control structure can be graphically represented as in Fig. 3, with the sensitivity-conditioning elements acting as a derivative-type control.

In compact matrix form, the closed-loop system reads as

\[
\dot{x} = T A x + B x^s, \quad x = [x_1^T, \zeta_1^T, x_2^T, \zeta_2^T]^T
\]

\[
A = \begin{bmatrix} a_1 & 0 & b_1 & 0 \\ 1 & 0 & 0 & 0 \\ -K_{P,2}a_1 + K_{P,1} & -K_{P,2}b_1 & K_{I,1}b_1 & -K_{P,2} \\ 0 & 0 & b_1 & 0 \end{bmatrix}
\]

\[
T = \begin{bmatrix} I & 0 \\ S^{\mathbf{g}}_{[x_2, \zeta_2]} & I \end{bmatrix}, \quad T^{-1} = \begin{bmatrix} I & 0 \\ -S^{\mathbf{g}}_{[x_2, \zeta_2]} & I \end{bmatrix}
\]

\[
B = \begin{bmatrix} 0 & -1 & K_{P,2} & -K_{P,1} \end{bmatrix}^T
\]

By means of the similarity transformation used in the proof of Proposition 2, we obtain

\[
T A \sim T^{-1}(T A) T = \begin{bmatrix} -K_{P,1} & -K_{I,1} & * & * \\ 1 & 0 & * & * \\ 0 & 0 & -K_{P,2} & -K_{I,2} \\ 0 & 0 & 0 & 1 \end{bmatrix}
\]

where $*$ are irrelevant terms for the following considerations. This block-companion form of $T A$ confirms that the only conditions required for stability of (20) are $K_{P,i} > 0$ and $K_{I,i} > 0$, as for (19). This is aligned with Proposition 2: Stability of (19) is preserved in (20) using the sensitivity conditioning (9). Note, that the system (20) without the sensitivity-conditioning term would have system matrix $A$ instead of $T A$: $\dot{x} = A x + B x^s$. Then, e.g., with $a_0 = 0, b_1 = 1, K_{P,1} = 1 > 0, K_{I,1} = 1 > 0, A$ has positive eigenvalues despite having positive control parameters, so stability is lost without the sensitivity-conditioning term.
Remark 6: In cascade control the steady-state closed form $x_2^k(x_1)$ in (7) is typically known by design. Hence, we could use an alternative sensitivity based on the Lyapunov function $\|x_2 - x_2^k(x_1)\|_{P_2}$, see Remark 4. Therefore, in this cascade control example (19), the sensitivity-conditioning approach (9) is equivalent to backstepping [11, Ch. 14] for nonlinear control design, up to an extra-proportional control term.

A. Numerical Simulation: DC/AC-Converter + RLC Filter

Designing a dc/ac-converter connected to an RLC-filter [33] is a standard control problem in power electronics. Here, we present a simplified version, where the dc/ac-converter modulates the dc voltage $v_{dc}$ into the three-phase ac voltage $v_m$. Using an averaged converter with stiff dc voltage, $v_m$ is a fully-controllable voltage source. This modulated voltage $v_m$ is then used to control the three-phase current $i$ through the resistance $R$ and inductance $L$, which in turn is used to control the output voltage $v$ at the capacitor $C$ to follow a reference $v^r$, see Fig. 4.

Let the electrical signals be represented in rectangular coordinates, using the real and imaginary parts, so $i, v, v_m \in \mathbb{R}^2$, and define the frequency $\omega$ and the rotation matrix $J = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$. Then, according to Kirchoff’s laws, the electrical signal dynamics in the rotation-frame coordinates [5] are

$$C \frac{dv}{dt} = i - J \omega C v, \quad L \frac{di}{dt} = v_m - (R + J \omega L)i - v.$$

Following the cascaded PI example (19), the reference $v^r$ and the controller $v_m$ are chosen as PI controllers with every $K(:,.) = k_l(:,\cdot)I_2$, where $I_2$ is identity of dimension 2

$$\frac{dv}{dt} = C^{-1}i - J \omega v,$$

$$\frac{d\xi}{dt} = (v - v^r),$$

$$i^r = J \omega C v + C(-K_{P,v}(v - v^r) - K_{I,v} \xi_v),$$

$$v_m = (R + J \omega L)i + v + L(-K_{P,i}(i - i^r) - K_{I,i} \xi_i),$$

$$\frac{di}{dt} = -K_{P,i}(i - i^r) - K_{I,i} \xi_i,$$

$$\frac{d\xi_i}{dt} = (i - i^r).$$

Fig. 4 shows simulation results for (21) with parameters in Table II. We simulate a black start, i.e., the system starts at time $t = 0$ with zero values, and needs to track a given constant reference, which can be interpreted as a step input. Since system (21) does not have an intrinsic time-scale separation, we use sufficiently large gains $k_{P,i}, k_{I,i}$ for the inner controller $C_2$ to artificially enforce a wider time-scale separation, i.e., lower $\epsilon$ in (3).

1) $k_{P,i} = 50 \frac{V}{\text{rad/sec}}, k_{I,i} = 100 \frac{V}{\text{rad}}$: the system without sensitivity conditioning is unstable even though the control parameters are all positive. On the contrary, the sensitivity conditioning turns it into a stable system, with magnitude $|v|$ stabilizing quickly at 1 p.u. and the frequency $f$ at 50 Hz.

2) $k_{P,i} = 100 \frac{V}{\text{rad/sec}}, k_{I,i} = 200 \frac{V}{\text{rad}}$: the system without sensitivity conditioning becomes stable, but it is still an unacceptably controller due to the high overshoot and relatively large settling time. Again, the sensitivity conditioning turns it into a well-performing controller.

3) $k_{P,i} = 250 \frac{V}{\text{rad/sec}}, k_{I,i} = 500 \frac{V}{\text{rad}}$: the system without sensitivity conditioning becomes stable with acceptable control performance. However, the sensitivity-conditioning approach performs much better with negligible overshoots.

Moreover, note in the zoom-in in Fig. 5 that the overshoot decreases as $k_{P,i}, k_{I,i}$ increase, so the system with sensitivity conditioning also benefits from having a faster controller $C_2$. Such faster $C_2$ can further increase the convergence rate.

V. Example II: Bilevel Optimization

In this section, we show an application of the sensitivity conditioning (9) to bilevel optimization [26]. As opposed to the cascade control example in Section IV, in this case, the steady-state map $x_2^k(x_1)$ is not available in closed form, thus backstepping is not applicable, see Remarks 4 and 6. Yet, the
sensitivity conditioning (9) can still be used to preserve the
stability of the two-time-scale system (7) in a single one.
Consider a general unconstrained bilevel problem [26, [34]
\[
\begin{align*}
\min_{x_1, x_2} & F_1(x_1, x_2) \\
\text{s.t.} & \quad x_2^* \in \arg \min_{x_2} F_2(x_1, x_2)
\end{align*}
\] (22)
where \( F_1(\cdot), F_2(\cdot) \) are the upper- and lower-level objective
functions, respectively.

Assumption 4 (Adaptation of Assumptions 1 and 2):
1) The functions \( F_1(\cdot) \) and \( F_2(\cdot) \) are twice and thrice con-
tinuously differentiable, respectively, with Lipschitz con-
tinuous partial derivatives.
2) For every \( x_1 \), the lower-level problem \( \arg \min_{x_2} F_2(x_1, x_2) \) has at most a single solution \( x_2^* \), where the
second-order partial derivative \( \nabla^2_{x_2 x_2} F_2(x_1, x_2^*) \) is inver-
tible.
The single-solution assumption is often used as simplification in bilevel problems [26]. It ensures that (22) is well-posed, and it allows to simplify the constraint to \( x_2 = \arg \min_{x_2} F_2(x_1, x_2) \). Moreover, the invertibility assumption allows to define the sensi-
tivity of \( x_2^* \), a known concept in bilevel optimization [35], similar to the one introduced in (6): Consider a point \( (x_1, x_2^*) \) satisfying the first-order optimality conditions of the lower-level problem, i.e., \( \nabla_{x_2} F_2(x_1, x_2^*) = 0 \). Since \( \nabla^2_{x_2 x_2} F_2(x_1, x_2^*) \) is invertible, the implicit-function theorem [28] guarantees the local existence of the map \( x_2^*(x_1) \), and gives an expression for its derivative
\[
\nabla_{x_1} x_2^*(x_1) = -\left( \nabla^2_{x_2 x_2} F_2(x_1, x_2^*(x_1)) \right)^{-1} \nabla_{x_2} F_2(x_1, x_2^*(x_1)).
\]
Additionally, \( x_2^*(x_1) \) can be used to locally define a reduced objective \( F_1^* : F_1^*(x_1) := F_1(x_1, x_2^*(x_1)) \) as in (7), and use \( \nabla_{x_1} x_2^*(x_1) \) to give an expression for the total derivative
\[
\begin{align*}
D_{x_1} F_1(x_1, x_2^*(x_1)) := & \nabla_{x_1} F_1^*(x_1) \\
& = \nabla_{x_1} F_1(x_1, x_2^*(x_1)) + \nabla_{x_1} x_2^*(x_1)^T \nabla_{x_2} F_2(x_1, x_2^*(x_1))
\end{align*}
\] (23)
defined for points where \( x_2 = x_2^*(x_1) \). Under Assumption 4, \( \nabla^2_{x_2 x_2} F_2(x_1, x_2) \) is invertible for \( x_2 \) in a neighborhood of \( x_2^*(x_1) \). Hence, \( \nabla_{x_1} x_2^*(x_1) \) and \( \nabla_{x_1} F_1^*(x_1) \) can be extended to these \( x_2 \), as the extended sensitivity in (6)
\[
\begin{align*}
D_{x_1} F_1(x_1, x_2) := & \nabla_{x_1} F_1(x_1, x_2) \\
& + S_{x_2^*}^2(x_1, x_2) \nabla_{x_2} F_1(x_1, x_2) \\
S_{x_2^*}^2(x_1, x_2) := & -\left( \nabla^2_{x_2 x_2} F_2(x_1, x_2) \right)^{-1} \nabla^2_{x_2 x_2} F_2(x_1, x_2)
\end{align*}
\] (24)
satisfying the restrictions \( D_{x_1} F_1(x_1, x_2) |_{(x_1, x_2^*(x_1))} = \nabla_{x_1} F_1^*(x_1) \), \( S_{x_2^*}^2(x_1, x_2) |_{(x_1, x_2^*(x_1))} = \nabla_{x_1} x_2^*(x_1) \).

A. Bilevel Local Solutions
Understanding the properties of the bilevel-problem solutions is essential to connect the convergence of algorithms with the stability of steady states from previous sections. Therefore, we
recall the concept of local solutions in [26, Ch. 8] to represent
local minima of (22) and their first- and second-order necessary
and sufficient conditions:

**Definition 1 (local solution):** [26, Ch. 8] A point \((x_1^*, x_2^*)\) is a (strict) local solution of (22) if.
1) The point \(x_2^*\) is a local minimum of \(F_2(x_1^*, \cdot)\) with fixed \(x_1^*\).
2) There exists a neighborhood \(N\) of \((x_1^*, x_2^*)\), such that,
\( F_1(x_1, x_2^*) \leq F_1(x_1, x_2) < \) for strict (for all \((x_1, x_2) \in N\), such that, \(x_2\) is a local minimum of \(F_2(x_1, \cdot)\) with fixed \(x_1\).

**Proposition 6 (First-order necessary conditions):** [26, Ch. 8] A local solution \((x_1^*, x_2^*)\) is a stationary point, i.e., it satisfies
\[
\nabla_{x_2} F_2(x_1^*, x_2^*) = 0, \quad \nabla_{x_1} F_1^*(x_1^*) = 0.
\]

**Proposition 7 (Second-order conditions):** [26, Ch. 8] Necessary conditions: A local solution \((x_1^*, x_2^*)\) satisfies
\[
\nabla^2_{x_2 x_2} F_2(x_1^*, x_2^*) \geq 0, \quad \nabla^2_{x_1 x_1} F_1^*(x_1^*) \geq 0.
\]

Sufficient conditions: A stationary point \((x_1^*, x_2^*)\) satisfying
\[
\nabla^2_{x_2 x_2} F_2(x_1^*, x_2^*) > 0, \quad \nabla^2_{x_1 x_1} F_1^*(x_1^*) > 0
\]
(25)
is a strict local solution.

B. Bilevel Gradient Flow
The steepest descent direction method [36] is a standard
approach to iteratively solve (22). It follows the negative gradient
of \(F_1^*(x_1)\) with step size \(\alpha^k\) in each iteration \(k\)
\[
\begin{align*}
x_1^{k+1} = & \quad x_1^k - \alpha^k \nabla_{x_1} F_1^*(x_1^k) = x_1^k - \alpha^k \left(D_{x_1} F_1(x_1^k, x_2^*)\right) \\
x_2^k = & \quad \arg \min_{x_2} F_2(x_1^k, x_2). \quad (26)
\end{align*}
\]
If the lower-level update \(x_2^k = \arg \min_{x_2} F_2(x_1^k, x_2)\) is not available in closed form, it can be solved iteratively using, e.g.,
gradient descent with step size \(\beta^k\) and updates
\[
x_1^{k+1} = x_1^k - \beta^k \nabla_{x_1} F_2(x_1^k, x_2^*)
\]
The corresponding continuous-time version of this bilevel gra-
dient descent (26) can be represented on two time scales with \(\epsilon \to 0\) and the singular perturbation interconnection (3)
\[
\begin{align*}
\dot{x}_1 = & \quad -D_{x_1} F_1(x_1, x_2) \\
\dot{x}_2 = & \quad -\nabla_{x_2} F_2(x_1, x_2). \quad (27)
\end{align*}
\]
As mentioned before in Section II, these nested iterations
(26) on two time scales (27) may slow down the algorithm
convergence. On the other hand, the sensitivity-conditioning system (9) yields
\[
\begin{align*}
\dot{x}_1 = & \quad -D_{x_1} F_1(x_1, x_2) \\
\dot{x}_2 = & \quad -\nabla_{x_2} F_2(x_1, x_2) + S_{x_2^*}^2(x_1, x_2) \dot{x}_1. \quad (28)
\end{align*}
\]
**Corollary 4 (Local convergence of (28)):** A point \((x_1^*, x_2^*)\) is a
strict local solution of (22) satisfying the sufficient conditions in (25), if and only if it is a locally exponentially stable steady state of the sensitivity-conditioning bilevel gradient flow (28).

**Proof:** Note, that Assumption 4 adapts Assumptions 1 and 2 for the bilevel problem (22), and the second-order total derivative
is symmetric and satisfies
\[
D_{x_1}^2 F_1(x_1, x_2^2(x_1)) := \nabla_{x_1}^2 F_1(x_1)
\]
\[
= \nabla_{x_1} D_{x_2} F_1(x_1, x_2^2(x_1))
\]
\[
+ \nabla_{x_2} x_2^2(x_1)^T \nabla_{x_2} D_{x_2} F_1(x_1, x_2^2(x_1)).
\]
Hence, \((x_1^*, x_2^*)\) is locally exponentially stable if and only if (25) holds [11, Corollary 4.3].

Remark 7: This convergence result can be stated to larger regions if similar conditions as in Proposition 3 hold.

C. Time Discretization and Numerical Simulation

The Euler-forward method [37] with time constant \(\tau\) can be used to integrate the differential equations (28) and (27) for a fixed \(\epsilon\). Then, we get a discrete-time descent algorithm
\[
x_{k+1}^1 = x_k^1 - \tau D_{x_1} F_1(x_k^1, x_k^2)
\]
\[
(27): x_{k+1}^2 = x_k^2 - \frac{\tau}{\epsilon} \nabla_{x_2} F_2(x_k^1, x_k^2)
\]
\[
(28): x_{k+1}^2 = x_k^2 - \tau \nabla_{x_2} F_2(x_k^1, x_k^2)
\]
\[
+ S_{x_2}^2(x_k^1, x_k^2)(-\tau D_{x_1} F_1(x_k^1, x_k^2))
\]
(29)
where the time constant \(\tau\) plays the role of the step size in optimization [16]. If (28) is locally or globally exponentially stable, see Propositions 2 and 3, then under some conditions its Euler-forward discretization will retain this exponential stability for suitable time constants below a certain threshold \(\tau < \tilde{\tau}\) [38], [39]. More concretely, this can be proven by extending [40, Lemma 5] to the Krassovski-Lyapunov functions in Proposition 3. We will formalize further results for discrete-time sensitivity conditioning in Subsection VI-C.

Remark 8: Bilevel-optimization problems like (22) can also be represented as Stackelberg games [41]. The singular perturbation dynamics (27), with its discrete-time version in (29), can be interpreted as a simultaneous gradient descent (\(\epsilon\)-GD) on both variables with different step sizes: \(\tau, \frac{\tau}{\epsilon}\). This corresponds to deterministic Stackelberg-learning dynamics [42], [43].

The particular case when \(F_2(x_1, x_2) = -F_1(x_1, x_2)\) in (22), is called a zero-sum or minimax game [42], [44]. In this context, this \(\epsilon\)-GD algorithm is known as the \(\gamma\)-gradient descent ascent (\(\gamma\)-GDA)[44], with \(\gamma = 1/\epsilon\). Then, the discrete-time sensitivity-conditioning application for bilevel optimization in (29), corresponds to the Stackelberg generalization of the algorithm in [23] for minimax games.

Consider an example similar to the ones in [23] and [42]
\[
F_1(x_1, x_2) = -\frac{x_1^2}{2} + x_2^2, \quad F_2(x_1, x_2) = \left(\frac{x_2^2}{4} - \frac{x_1 x_2}{2}\right) e^{-\frac{x_2^2}{2}}.
\]
(30)
The point \((0, 0)\) is a strict local solution of the bilevel problem (22), since \(\nabla_{x_2} F_2(x_1, x_2) = 0, D_{x_2} F_1(x_1, x_2) = 0, \nabla_{x_2}^2 F_2(x_1, x_2) = 1/2 > 0, D_{x_2}^2 F_1(x_1, x_2) = 1 > 0\). In Fig. 6, we compare the approaches in (29) using \(\tau = 1/4\), singular-perturbation values \(\epsilon \in \{1/4, 1/2\}\), and the functions in (30). First, we observe that the sensitivity-conditioning (P-S) approach (28) is able to converge quickly to the solution \((0, 0)\). For the case \(\epsilon = 1/2\), the \(\epsilon\)-GD based on (27) fails to converge.

It converges for \(\epsilon = \frac{1}{4}\), but still oscillates around the solution, causing a slower convergence.

VI. SENSITIVITY CONDITIONING FOR MULTIPLE SYSTEMS

The theoretical results in Section III, and the applications to cascade control and bilevel optimization, Sections IV and V respectively, deal with two-time-scale systems. In this section, we show how the sensitivity-conditioning interconnection (9) can be extended to multiple time scales arising in, e.g., multiple nested systems [12] or multilevel programming [45]. Consider \(N\) differential-algebraic-equation subsystems \(\Sigma_i\), with states \(x_i \in \mathbb{R}^{n_i}\) and vector fields \(f_i(\cdot)\), operating on different time scales \(\tau_i\) as in (2), ordered from slow \(\Sigma_1\) to fast \(\Sigma_N\)
\[
\frac{dx_i}{d\tau_i} = f_i(x_1, \ldots, x_N) \text{ s.t. } f_j(x_1, \ldots, x_N) = 0 \forall j > i
\]
\[
\frac{dx_i}{d\tau_i} = 0 \forall j < i.
\]
(31)
As in (3), consider the corresponding singular-perturbed system with terms \(0 < \epsilon_N \ll \cdots \ll \epsilon_2 \ll \epsilon_1 = 1\) [12]
\[
\begin{bmatrix}
\epsilon_1 I & 0 & 0 \\
0 & \ddots & 0 \\
0 & 0 & \epsilon_N I
\end{bmatrix}
\begin{bmatrix}
x_1 \\
\vdots \\
x_N
\end{bmatrix}
= \begin{bmatrix}
f_1(x_1, \ldots, x_N) \\
\vdots \\
f_N(x_1, \ldots, x_N)
\end{bmatrix}
\]
where the time-scale separation in (31) is recovered for \(\tau_i = \frac{1}{\epsilon_i}\) in the singular limit \(\epsilon_{i+1}/\epsilon_i \to 0 \forall i = 1, \ldots, N-1\).

A. Steady States, Sensitivities, and Total Derivatives

Assume that every subsystem \(\Sigma_i\) in (31) has isolated steady states, and that the implicit-function theorem [28] can be used to guarantee the existence of steady-state maps \(x_i^s(x_1, \ldots, x_{i-1})\) recursively from fast to slow subsystems \(\Sigma_i\): First, for some \(x_N\), such that, \(0 = f_N(x_1, \ldots, x_N)\) for the fastest \(\Sigma_N\), the implicit-function theorem guarantees the local existence of \(x_N^s(x_1, \ldots, x_{N-1})\). Under time-scale separation, the next system \(\Sigma_{N-1}\) has reduced-order dynamics \(x_{N-1} = f_{N-1}(x_1, \ldots, x_{N-1}) := f_{N-1}(x_1, \ldots, x_{N-1}, x_N^s(x_1, \ldots, x_{N-1}))\), which allow now to define the steady-state map \(x_{N-1}^s(x_1, \ldots, x_{N-2})\), and recursively \(x_{i-1}^s(x_1, \ldots, x_{i-1})\), which depend only on the states of the slower systems \(x_1, \ldots, x_{i-1}\). To ease the notation, from now on we will use \(x_i^s\) instead of \(x_i^s(x_1, \ldots, x_{i-1})\) to denote the steady-state map and
its dependencies. Similarly, we use \((x_1, \ldots, x_i, x_{i+1}^+, \ldots, x_N^+)\) to denote that \(x_{i+1}^+, \ldots, x_N^+\) are all at steady state for given values \(x_1, \ldots, x_i\). As in (7), the steady-state maps allow to define the reduced-order dynamics

\[
\frac{dx_i}{dt} = f_i^c(x_1, \ldots, x_i) := f_i(x_1, \ldots, x_i, x_{i+1}^+, \ldots, x_N^+). \tag{32}
\]

Under sufficient regularity, the implicit-function theorem gives the sensitivity of each steady state \(x_i^s\) with respect to any \(x_j\) for \(j < i\)

\[
\nabla x_j x_i^s(x_1, \ldots, x_{i-1}) = \left(-\nabla x_j f_i^c(x_1, \ldots, x_{i-1}, x_i^+)^{-1}\nabla x_j f_i^c(x_1, \ldots, x_{i-1}, x_i^+)\right).
\tag{33}
\]

Now, the concepts of extended sensitivities (6) and extended total derivatives (24) can be used to define compact analytical expressions for these sensitivities \(\nabla x_j x_i^s\) and an extension for a general point \((x_1, \ldots, x_N)\). We define the extended total derivatives and sensitivities for every \(i\) and \(j\) recursively from \(N\) to 1

\[
D_{x_N} f_i := \nabla x_N f_i
\]

\[
D_{x_j} f_i := \nabla x_j f_i + \sum_{k = \text{max}(i,j) + 1}^N D_{x_k} f_i S_{x_j} x_k
\]

\[
S_{x_j} x_k := -(D_{x_j} f_i)^{-1} D_{x_j} f_i \quad \forall j < i \tag{34}
\]

where for clarity we omit the evaluation at \((x_1, \ldots, x_N)\). These extended total derivatives of \(f_i\) with respect to \(x_j\) take into account the dependency of each intermediate \(x_k^+, k \geq \text{max}(i, j) + 1\), with respect to \(x_j\). When restricted, they coincide with the total derivatives (24) of \(f_i^s(x_1, \ldots, x_i)\), and the steady-state sensitivities (33)

\[
D_{x_j} f_i(x_1, \ldots, x_N)_{|_{x_1, \ldots, x_i, x_{i+1}^+, \ldots, x_N^+}} = \nabla x_j f_i^c(x_1, \ldots, x_i)
\]

\[
S_{x_j} x_k(x_1, \ldots, x_N)_{|_{x_1, \ldots, x_i, x_{i+1}^+, \ldots, x_N^+}} = \nabla x_j x_i^s(x_1, \ldots, x_{i-1}). \tag{35}
\]

To guarantee that the implicit-function theorem is applicable for every \(f_i(x_1, \ldots, x_i, x_{i+1}^+, \ldots, x_N^+) = 0\) and, thus, ensure that steady states \(x_i^s\), reduced-order dynamics (32), sensitivities (III-A), and extended sensitivities and total derivatives (34), are well defined around steady states, we formalize the assumptions made in this section in the following one:

**Assumption 5 (Extension of Assumption 1):** For all \(i > 1\), the vector fields \(f_i(\cdot)\), \(f_i^s(\cdot)\) are continuously differentiable, and the reduced-order systems (32) have isolated steady states \(x_i^s\), where the partial derivatives \(\nabla x_j f_i^c(x_1, \ldots, x_{i-1}, x_i^+)^r\) are invertible.

This assumption also implies that the total derivatives \(D_{x_j} f_i(\cdot)\) in (34) are invertible for \((x_1, \ldots, x_N)\) in a neighborhood of \((x_i^s, \ldots, x_N^+)\). Note, that in contrast to Assumption 1, Assumption 5 relaxes the need for a single steady state in each subsystem, since there may exist multiple steady states \(x_i^s\) for every subsystem \(\Sigma_i\), given the values \(x_1, \ldots, x_{i-1}\), i.e., the set of steady states

\[
\mathcal{X}_i^s(x_1, \ldots, x_{i-1}) := \{x_i| f_i(x_1, \ldots, x_i, x_{i+1}^+, \ldots, x_N^+) = 0\}
\]

is not necessarily a singleton. However, given the locally invertible \(D_{x_j} f_i(\cdot)\), steady states will be isolated points. Moreover, given the implicit-function theorem, the extended sensitivity \(S_{x_j} x_i^s(x_1, \ldots, x_{i-1}, x_i^+, \ldots, x_N^+)\) (34) evaluated at each \(x_i^s \in \mathcal{X}_i^s\) gives the actual sensitivity of each \(x_i^s\) with respect to \(x_1, \ldots, x_{i-1}\). Therefore, Theorem 1 presented later will allow to test the local stability of every combination of steady states \((x_1^s, \ldots, x_N^s)\), where \(x_i^s \in \mathcal{X}_i^s\) for every \(i\).

### B. Sensitivity Conditioning for Multiple Time Scales

With the previously defined extended total derivatives and sensitivities (34), the sensitivity-conditioning system (9) from Section III can be extended and applied to the multiple-timescales system (31)

\[
\begin{bmatrix}
I & 0 & 0 & 0 \\
-S_{x_1}^2 & I & 0 & 0 \\
\vdots & \ddots & \ddots & \vdots \\
-S_{x_N} & \ldots & -S_{x_N}^2 & I
\end{bmatrix}
\begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2 \\
\vdots \\
\dot{x}_N
\end{bmatrix}
= \begin{bmatrix}
f_1(x_1, \ldots, x_N) \\
f_2(x_1, \ldots, x_N) \\
\vdots \\
f_N(x_1, \ldots, x_N)
\end{bmatrix}
\tag{36}
\]

where for clarity we omit the evaluation at point \((x_1, \ldots, x_N)\) in the conditioning matrix \(M\). In the equivalent expression

\[
\Sigma_i : \dot{x}_i = f_i(x_1, \ldots, x_N) + \sum_{j=1}^{i-1} S_{x_j} x_i^s(x_1, \ldots, x_N) \dot{x}_j
\tag{37}
\]

the extra terms \(\sum_{j=1}^{i-1} S_{x_j} x_i^s(x_1, \ldots, x_N) \dot{x}_j\) play again the role of predicting and anticipating the changes of steady states \(x_i^s\) due to the slower dynamics \(\dot{x}_j\), \(j < i\). Note that our approach (36) does not require any specific cascaded structure.

As in Assumption 2, to guarantee local existence and uniqueness [11, Th. 3.1] of a solution for (36), we assume:

**Assumption 6:** The vector fields in (36) and (37) are locally Lipschitz continuous.

Then, the statements in Propositions 1 to 3 for the two-systems sensitivity conditioning (9), can be extended to the multiple-timescale case sensitivity conditioning (36):

**Theorem 1 (Extension of Propositions 1, 2, and Corollary 2):**

1. **Positive invariance:** For some \(p \geq 1\), given the dynamics (37) of \(\dot{x}_i\) for \(i \geq p\), initialized at time \(t_0\): \(x_i(t_0) = x_i^s(x_1(t_0), \ldots, x_{i-1}(t_0))\). Then, \(x_i(t) = x_i^s(x_i(t), \ldots, x_{i-1}(t))\) \(\forall t \geq p\) is the unique solution on the open domain of existence.

2. **Local stability:** At a steady state \((x_1^s, \ldots, x_N^s)\) the Jacobian \(J\) of (36) satisfies

\[
J = \begin{bmatrix}
\nabla x_1 f_1^c(x_1^s) & * & * \\
0 & \ddots & * \\
0 & 0 & \nabla x_N f_N(x_1^s, \ldots, x_N^s)
\end{bmatrix},
\tag{38}
\]

where * are irrelevant terms.

3. **Global exponential stability:** Assume that the vector fields \(f_i(\cdot)\) are globally Lipschitz continuous, that the total derivatives \(D_{x_j} f_i(\cdot)\) in (34) are globally invertible, and
that there exists positive-definite matrices $P_i > 0$ and $\eta_i$, such that, for all $(x_1, \ldots, x_N)$, the following contraction condition holds:\footnote{This would correspond to the condition $P_i \nabla x_i f^T_i + (\nabla x_i f^T_i) P_i^T \preceq -\eta_i P_i$ in Corollary 2 for the two-time-scale system. However, here, it is required to hold for an extended number of points $(x_1, \ldots, x_N)$, not just for only $(x_1, \ldots, x_{i+1}, \ldots, x_N)$, hence, it is more strict.}

$$P_i D x_i f_i + D x_i f^T_i P_i^T \leq -\eta_i P_i. \quad (39)$$

Then, there exists a unique steady state $(x^*_1, \ldots, x^*_N)$, which is a globally exponentially stable steady state of the sensitivity-conditioning approach (36).

**Proof:** See Appendix B. $\blacksquare$

**Corollary 5 (of Theorem 1.2, extending Corollary 1):** The Jacobians (35) and (38), for the systems under time-scale separation (32) and the sensitivity conditioning (36), respectively, have the same eigenvalues and, thus, the same local stability.

The multiple time-scale sensitivity conditioning (36) can also be generalized as (16), to improve the performance. Theorem 1 can then be extended as in Propositions 4 and 5.

### C. Discrete-Time Sensitivity Conditioning

Here, we show how the multiple time-scale sensitivity conditioning (36) can be extended to discrete-time systems, while preserving the local stability result in Theorem 1. Consider the discrete-time systems

$$\Sigma_i: x_i^{k+1} = x_i^k + f_i(x_i^1, \ldots, x_i^k)$$

where $x_i^k$ denotes the value of $x_i$ at time $t^k$. Under-time scale separation as in (32), we can represent each discrete-time subsystem $\Sigma_i$ in its own time scale $k_i$ with reduced-order dynamics

$$x_i^{k+1} = x_i^k + f_i(x_i^1, \ldots, x_{i-1}^k, x_i^k)$$

$$\vdots$$

where $x_i^{k+1} = x_i^k$ for $j < i$, and $x_i^j$ is the steady state of $x_j$ given $x_1^1, \ldots, x_{i-1}^k, x_i^k, x_{i+1}^k, \ldots, x_N^k$, for $j > i$. Similar to (35), the Jacobians of (40) are

$$J_i \equiv I + \nabla x_i f^T_i (x_1^1, \ldots, x_i^k) \equiv I + D x_i f_i (x_1^1, \ldots, x_N^k)$$

and, thus, systems (40) are locally asymptotically stable if the eigenvalues $\lambda_i$ of $I + \nabla x_i f^T_i$ satisfy $|\lambda_i| < 1$. Using the same extended sensitivities $S^*_{x_i}$ as in (34), the discrete-time version of the sensitivity conditioning (36) can be expressed as

$$\begin{bmatrix}
I & 0 & 0 & 0 \\
-S_{x_1} & I & 0 & 0 \\
\vdots & \ddots & \ddots & \ddots \\
-S_{x_N} & \ldots & -S_{x_{N-1}} & I \\
\end{bmatrix}
\begin{bmatrix}
x_1^{k+1} - x_1^k \\
s_{x_1} \\
\vdots \\
s_{x_N} \\
\end{bmatrix}
= \begin{bmatrix}
f_1 \\
\vdots \\
f_N \\
\end{bmatrix} \quad (41)$$

omitting for clarity the evaluation at the point $(x^*_1, \ldots, x^*_N)$.

**Proposition 8:** At the steady state $(x^*_1, \ldots, x^*_N)$ the Jacobians of (40) and (41) have the same eigenvalues. Thus, (40) is locally exponentially stable if and only if (41) is so. Moreover, if for some $i$ the Jacobian $I + \nabla x_i f^T_i$ has any eigenvalue with norm larger than one, then both (40) and (41) are unstable.

**Proof:** The proof follows similar steps as the local stability one in Theorem 1: After performing the same similarity transformations, the Jacobian $J$ of (41) satisfies

$$J \sim I + \begin{bmatrix}
\nabla x_1 f^T_i (x^*_1) & * & * \\
0 & \ddots & * \\
0 & 0 & \nabla x_N f^T_i (x^*_1, \ldots, x^*_N) \\
\end{bmatrix}.$$ $\blacksquare$

### VII. Conclusion and Outlook

In this work, we have presented the sensitivity conditioning: An alternative design tool for interconnected systems, that uses a predictive feed-forward term to preserve the stability of the system analyzed at different time scales. This approach does not introduce a lower threshold on the actual time-scale separation between subsystems, in contrast to the usual singular perturbation approach. Moreover, we have shown examples of control-design problems and optimization algorithms where our approach can be directly applied and improves the performance compared to a time-scale-separation approach.

We believe that the applicability of our approach is not limited to these examples, but has the potential to be used in many other applications. For example, for any nested algorithms (e.g., in optimization or adaptive control) this sensitivity conditioning could be used to design faster algorithms avoiding the need of time-scale separation between nested iterations. This is particularly promising for cases where iterations are computationally expensive, even when they are simple to evaluate, e.g., in distributed algorithms with communication bottlenecks.

Several directions for future research remain open: Since the sensitivities employed in the proposed conditioning are heavily model-based, we have established input-to-state stability robustness analysis against model errors. Nonetheless, a more sophisticated, robust performance guarantee would be desirable. Moreover, through this work, we have considered only continuously-differentiable vector fields driving the dynamics. Thus, it remains to be seen how this method could be extended to nondifferentiable cases, arising often in optimization.

### APPENDIX A

**Proof of Lemma 1**

For clarity, we omit the evaluation at a given $x$. Consider the singular value decomposition $\nabla x f = USV^T$. Let $\sigma_{\min} = \min_i S_{x,i}$ denote the minimum singular value of $\nabla x f$, and $v_{\min}, u_{\min}$ the columns of $V, U$ corresponding to $\sigma_{\min}$. Since $P > 0$ and $\|v_{\min}\|_2 = \|u_{\min}\|_2 = 1$, we have

$$0 < \eta_{\sigma_{\min}}(P) \leq \eta_{\sigma_{\min}}^T P \eta_{\sigma_{\min}}$$

$$\leq -\sigma_{\min}^T (P \nabla x f + \nabla x f^T P) v_{\min}$$

$$= -2 \sigma_{\min} v_{\min}^T P u_{\min}$$

$$\leq 2 \sigma_{\min} \|v_{\min}\|_2 \|u_{\min}\|_2 \lambda_{\max}(P) = 2 \sigma_{\min} \lambda_{\max}(P)$$
where the third inequality is due to $P \nabla_x f(x) + \nabla_x f(x)^T P \preceq -\eta P \forall x$. Thus, $\nabla_x f^{-1} = V S^{-1} U^T$ is well defined and 

$$\| \nabla_x f^{-1} \|_2 = \max_{i} \frac{1}{s_{\min}} = \frac{\sigma_{\min}(P)}{\sigma_{\min}(P)} \leq \frac{\sigma_{\max}(P)}{\sigma_{\min}(P)}.$$ 

Next, with $x(\delta) = x^s + \delta(x - x^s)$ for $\delta \in [0,1]$, then 

$$(x - x^s)^T P (f(x) - f(x^s)) + (f(x) - f(x^s))^T P (x - x^s) = \int_0^1 (x - x^s)^T P \nabla_x f(x(\delta)) + \nabla_x f(x(\delta)^T) P (x - x^s) \, d\delta \leq -\eta \|x - x^s\|_2^2 \leq 0,$$

where the first inequality holds because $P \nabla_x f(x(\delta)) + \nabla_x f(x(\delta)^T) P \preceq -\eta P$ for $x(\delta) \in B_r(x^s)$. Hence, $2\|f(x)\|_2^2 = 2\|f(x)\|_P\|x - x^s\|_P \geq \eta \|x - x^s\|_2^2$ and $\|f(x)\|_P \geq \frac{\eta}{2} \|x - x^s\|_P$.

### APPENDIX B

**Proof of Theorem 1**

1) For any $i \geq p$, consider the dynamics of $\dot{x}_i$ in (37), initialized at time $t_0$ as $x_i(t_0) = x^*_i(x_1(t_0), \ldots, x_{i-1}(t_0))$. Local existence and uniqueness of a guarantee is provided by Assumption 6. Then, $x_i(t) = x^*_i(x(t), \ldots, x_{i-1}(t))$ is the unique solution on the open domain of existence, since the derivative 

$$\dot{x}_i(t) = f_i(x_1(t), \ldots, x_{i-1}(t), x_i(t), \ldots, x_N(t))$$

$$+ \sum_{j=1}^{i-1} S_{x_j x_i}(x_1(t), \ldots, x_{i-1}(t), x_i(t), \ldots, x_N(t)) \dot{x}_j(t)$$

$$= \frac{dx^*_i(x_1(t), \ldots, x_{i-1}(t))}{dt}$$

and the initial conditions coincide for $t \geq t_0$ on the domain of existence.

2) By recursively using the expression in (37), the sensitivity-conditioning system (36) can be rewritten as 

$$\begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_N \end{bmatrix} = \begin{bmatrix} I & 0 \\ \vdots & I \\ \vdots & \vdots \\ I & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} + \begin{bmatrix} -S_{x_1 x_2}(I) & \cdots & \cdots & \vdots \\ \vdots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ -S_{x_N x_1}(I) & \cdots & \cdots & I \end{bmatrix} \begin{bmatrix} f_1(x_1(t), \ldots, x_{i-1}(t)) \\ \vdots \\ f_N(x_1(t), \ldots, x_{i-1}(t)) \end{bmatrix}$$

where the matrices $[S_{x_1 x_2}(I), \ldots, S_{x_N x_1}(I)]$ are the row block $i$. Since at a steady state $(x^*_1, \ldots, x^*_N)$ we have $f_i(x^*_1, \ldots, x^*_N) = 0$, then the Jacobian of (36) at $(x^*_1, \ldots, x^*_N)$ simplifies to 

$$J = \begin{bmatrix} I \\ \vdots \\ I \end{bmatrix} \begin{bmatrix} -S_{x_1 x_2}(I) & \cdots & \cdots & -S_{x_N x_1}(I) \end{bmatrix} \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \begin{bmatrix} f_1(x^*_1, \ldots, x^*_N) \\ \vdots \\ f_N(x^*_1, \ldots, x^*_N) \end{bmatrix}$$

where the $i,j$-block of $\nabla_x f$ is $(\nabla_x f)_{ij} = \nabla_{x_j} f_i$. 

Since $\begin{bmatrix} I \\ \vdots \\ I \end{bmatrix} \begin{bmatrix} -S_{x_1 x_2}(I) & \cdots & \cdots & -S_{x_N x_1}(I) \end{bmatrix} = \begin{bmatrix} -S_{x_1 x_2}(I) & \cdots & \cdots & -S_{x_N x_1}(I) \end{bmatrix} \begin{bmatrix} I \\ \vdots \\ I \end{bmatrix}$, multiplying $J$ on both sides by these matrices we can iteratively construct matrices similar to $J$, starting with $\tilde{J}_N = J$, and sequentially defining the matrices $\tilde{J}_i$ as 

$$\tilde{J}_i = \begin{bmatrix} S_{x_1 x_2}^i & \cdots & \cdots & S_{x_N x_1}^i \\ 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & I \end{bmatrix} \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \cdots & \ddots & \vdots \\ 0 & \cdots & \cdots & 1 \end{bmatrix} \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}$$

It can be shown that lower diagonal in $\tilde{J}_i$ terms cancel until we have 

$$J = \tilde{J}_N \sim \tilde{J}_{N-1} \sim \cdots \sim \tilde{J}_1 = \begin{bmatrix} D_{x_1 f_1} & * & * \\ 0 & \cdots & \cdots \\ 0 & \cdots & \cdots \\ D_{x_N f_N} & \cdots & \cdots \end{bmatrix},$$

and at steady-state $D_{x_i} f_i = \nabla_{x_i} f_i^T \forall i$.

3) There is a unique steady state as a consequence of the contraction condition (39) and Lemma 1. First, consider the matrix with sensitivities in (36) truncated at $i$:

$$S_i = \begin{bmatrix} I & 0 & 0 \\ -S_{x_1 x_2} & \cdots & \cdots \\ \vdots & \ddots & \ddots \\ -S_{x_N x_1} & \cdots & \cdots \\ I & 0 & \cdots \end{bmatrix}$$

then using (37) and the definitions in (34) we have 

$$\begin{bmatrix} \nabla_{x_1} f_i \\ \vdots \\ \nabla_{x_N} f_i \end{bmatrix} = \nabla_{x_i} f_i^T \begin{bmatrix} \hat{x}_1 \\ \vdots \\ \hat{x}_N \end{bmatrix}^T$$

$$= \begin{bmatrix} D_{x_1} f_i & \cdots & D_{x_N} f_i \end{bmatrix} S_i \begin{bmatrix} \hat{x}_1 \\ \vdots \\ \hat{x}_N \end{bmatrix}^T$$

$$= \begin{bmatrix} D_{x_1} f_i & \cdots & D_{x_N} f_i \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \vdots \\ \hat{x}_i \\ f_{i+1} \\ \cdots \end{bmatrix}^T$$

$$= \sum_{j=1}^{i-1} D_{x_j} f_i \hat{x}_j + D_{x_i} f_i (f_i + \sum_{j=1}^{i-1} S_{x_j x_i}^j) + \sum_{j=i+1}^{N} D_{x_j} f_i f_j$$

$$= \sum_{j=1}^{N} D_{x_j} f_i f_j$$

As in the proof of Proposition 3, let $L_{f_i}$ denote the Lipschitz constant of $f_i$. Given Lemma 1, it can be certified that the extended sensitivities $S_{x_i}^j$ and total derivatives $D_{x_i} f_i$ are all bounded, so there exists $L_{D_{x_i} f_i, x_j} > 0$ such that, $\|D_{x_i} f_i\|_2 \leq L_{D_{x_i} f_i, x_j}$. Defining the Lyapunov functions $V_i(x_1, \ldots, x_N) = \|f_i(x_1, \ldots, x_N)\|_P^2$, then we have 

$$\dot{V}_i = f_i^T P_i \begin{bmatrix} \nabla_{x_1} f_i \\ \vdots \\ \nabla_{x_N} f_i \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \vdots \\ \hat{x}_N \end{bmatrix}^T$$

$$+ \text{(transpose terms)}$$

$$= \sum_{j=1}^{N} D_{x_j} f_i P_i (D_{x_j} f_i f_j) + (f_i^T P_i (D_{x_j} f_i f_j))^T$$

$$\leq -\eta \|f_i\|_P + 2 \sum_{j=1}^{N} L_{D_{x_j} f_i, x_j} \|f_i\|_P \|f_j\|_P \|D_{x_j} f_i\|_{\sigma_{\min}(P)}$$

Consider the parameters $\theta_i > 0$, and the Lyapunov function $V(x_1, \ldots, x_N) = \sum_i \theta_i V_i(x_1, \ldots, x_N)$. Then, we have 

$$\dot{V} \leq -\|f_N\|_P \cdots \|f_i\|_P$$

$$W \begin{bmatrix} \|f_N\|_P \\ \vdots \end{bmatrix}^T$$
Let $W_i$ denote the upper-left minor $i$ of $W$. The first minor satisfies $W_1 = \mathbf{q}_1 := \theta_{NN} > 0$. We continue by recursion: Assume there exists $\mathbf{q}_{i-1} > 0$ so that $W_{i-1} \succeq \mathbf{q}_{i-1} I$, then $W_i > 0$ if

$$\theta_{i-1} \eta_i - \theta_i^2 \leq \frac{1}{\mathbf{q}_{i-1}} \sum_{j=1}^{N} \mathbf{v}^2_{i,j} > 0$$

or equivalently $\theta_i < \frac{\mathbf{q}_{i-1}}{\sum_{j=1}^{N} \mathbf{v}^2_{i,j}}$, which provides a recursive set of conditions on $\theta_i$ so that all minors $W_i$ are positive definite. In the end, $W_N = W > 0$, so there exists $\mathbf{q}_N > 0$ such that $W \geq \mathbf{q}_N I$ and

$$V \leq -\mathbf{q}_N \sum_{i=1}^{N} V_i \leq -\frac{\mathbf{q}_N}{\max_i \theta_i} V.$$
[42] T. Fiez, B. Chasnov, and L. J. Ratliff, “Convergence of learning dynamics in Stackelberg games,” 2019, arXiv:1906.01217.
[43] T. Fiez, B. Chasnov, and L. Ratliff, “Implicit learning dynamics in Stackelberg games: Equilibria characterization, convergence analysis, and empirical study,” in Proc. Int. Conf. Mach. Learn., 2020, pp. 3133–3144.
[44] C. Jin, P. Netrapalli, and M. Jordan, “What is local optimality in nonconvex-nonconcave minimax optimization?,” in Proc. Int. Conf. Mach. Learn., 2020, pp. 4880–4889.
[45] J. F. Bard and J. E. Falk, “An explicit solution to the multi-level programming problem,” Comput. Oper. Res., vol. 9, no. 1, pp. 77–100, 1982.

Miguel Picallo received two Diploma degrees in mathematics and industrial engineering from the Polytechnic University of Catalonia (UPC), Barcelona, Spain, in 2012, focusing on control theory, and the M.Sc. degree in management science and engineering from Stanford University, Stanford, CA, USA, in 2013, focusing on optimization and machine learning. He is currently working toward the Ph.D. degree in information technology and electrical engineering with the Automatic Control Laboratory, ETH Zurich, Zurich, Switzerland, under the supervision of Florian Dörfler. From 2013 to 2016, he was a Data Scientist with Accenture Analytics, and later founded Arcvi, a data analytics consultancy firm. His research interests include estimation, control, and real-time optimization with applications to electrical power grids.

Saverio Bolognani received the B.S. degree in information engineering, the M.S. degree in automation engineering, and the Ph.D. degree in information engineering from the University of Padua, Padua, Italy, in 2005, 2007, and 2011, respectively. From 2006 to 2007, he was a Visiting Graduate Student with the University of California at San Diego, San Diego, CA, USA. From 2013 to 2014, he was a Postdoctoral Associate with the Laboratory for Information and Decision Systems of the Massachusetts Institute of Technology in Cambridge, Cambridge, MA, USA. He is currently a Senior Scientist with the Automatic Control Laboratory, ETH Zurich, Zurich, Switzerland. His research interests include the application of networked control system theory to power systems, cyber-physical systems, the intersection of nonlinear optimization with feedback control design, multiagent systems, and game theory.

Florian Dörfler received the Diploma degree in engineering cybernetics from the University of Stuttgart, Stuttgart, Germany, in 2008, and the Ph.D. degree in mechanical engineering from the University of California at Santa Barbara, Santa Barbara, CA, USA, in 2013. From 2013 to 2014, he was an Assistant Professor with the University of California Los Angeles, Los Angeles, CA, USA. He is currently an Associate Professor with the Automatic Control Laboratory, ETH Zurich, Zurich, Switzerland, and the Associate Head of the Department of Information Technology and Electrical Engineering. His primary research interests include control, optimization, and system theory with applications in network systems, in particular electric power grids. Dr. Dörfler is a recipient of the Distinguished Young Research awards by International Federation of Automatic Control (IFAC) (Manfred Thoma Medal 2020) and European Control Award (EUCA) 2020. His students were winners or finalists for Best Student Paper awards at the European Control Conference (2013, 2019), the American Control Conference (2016), the Conference on Decision and Control (2020), the Conference on Intelligent Transportation Systems (2021), the Power and Energy Society (PES) General Meeting (2020), and the PES PowerTech Conference (2017). He is furthermore a recipient of the 2010 Automatic Control Council (ACC) Student Best Paper Award, the 2011 O. Hugo Schuck Best Paper Award, the 2012–2014 Automatica Best Paper Award, the 2016 IEEE Circuits and Systems Guillemin–Cauer Best Paper Award, and the 2015 University of California at Santa Barbara (UCSB) Mechanical Engineering (ME) Best Ph.D. award.