Optimization methods for dynamical systems learning

Simone Pirrera

April 7, 2025

1 Abstract

System identification (SI) is the discipline of learning dynamical systems from experimental data. This thesis considers the simulation error minimization (SEM) approach to SI and aims to develop algorithms for its solution. In this context, we address the vanishing and exploding gradients issues and propose a constrained optimization approach to overcome these challenges.

To solve the formulated constrained optimization problem, we concentrate on developing novel, efficient, first-order algorithms. To this end, we introduce the controlled multipliers optimization (CMO) framework, which reformulates the optimization problems as control problems, allowing us to utilize controller design to derive solutions. We present three distinct algorithms and analyze their stability.

Ultimately, we apply one of the proposed algorithms to SEM-based identification. Our approach's efficacy is validated through various benchmark tests and simulation experiments, revealing significant improvements in SI tasks, including black-box SI through artificial neural networks, and gray-box problems.

2 Summary of the thesis

System identification (SI) is the science of building mathematical models that describe physical or artificial systems and phenomena.
The general paradigm of SI is as follows. Firstly, we collect experimental data on the system; usually, these data are affected by noise. Secondly, we select a model class, i.e., a family of models to which we assume the system belongs. Then, we estimate the model's parameters by solving a suitably defined optimization problem. Finally, we validate the model using a data set not used during the estimation.

SI is crucial in various engineering applications. Depending on the context, the identified model is used for prediction, simulation [1, 2], controller design [3, 4], decision-making, and data filtering and denoising [5], among many others. Another increasingly widespread application of SI is direct data-driven control (DDDC) design. A popular approach to DDDC is recasting the controller design problem into the problem of identifying the controller directly from data, given performance specifications. Examples of DDDC methods following this approach are [6, 7, 8].

Most approaches to SI are based on the solution to a suitable optimization problem aiming to minimize prediction or simulation error. Mathematically, this consists of the solution to the optimization problem

$$\min_{\theta \in \mathbb{R}^{n_\theta}} \mathcal{L}(\theta | u, \tilde{y}), \tag{1}$$

where $u_t \in \mathbb{R}^q, \tilde{y}_t \in \mathbb{R}^p$ are the available input and output data, respectively, and $\mathcal{L} : \mathbb{R}^{n_\theta} \to \mathbb{R}$ accounts for the error. The definition of the cost function $\mathcal{L}$ depends on several aspects. Firstly, it depends on the considered error model and the norm used to assess it. Secondly, it depends on the considered model class. Finally, we can use different regularization terms to penalize the model's complexity.

There are several options available when it comes to selecting a model class. First, we can classify models as either state-space or input-output models. Next, we can differentiate models based on their structure, i.e., the functional relationship they define between input and output. The simplest structure is that of linearly parameterized models, where the model is a linear combination of specified functions. Alternatively, the model structure can be determined from physical equations; in this case, we deal with gray-box models. Another popular approach to defining the model structure is using flexible

function approximators, such as neural networks. The latter approach strongly connects classical SI and modern machine learning. Indeed, objects like recurrent neural networks are particular nonlinear state-space models. In this context, the terms "learn" and "identify" are synonymous.

Most approaches to SI rely on prediction error minimization (PEM). According to PEM, we identify the model parameters by minimizing the difference between the measured output and the one-step-ahead prediction provided by the model. Considering linearly parametrized models, the PEM problem is convex and can be solved efficiently. Instances of problems of this kind are, e.g., ARX identification [1] and kernel methods [9]. Instead, when the model structure is nonlinear in the parameters, the identification problem (1) is non-convex, and a tractable solution often requires looking for local minimizers. Moreover, PEM methods enjoy consistency (i.e., convergence of the estimate to the actual parameter value as the number of data tends to infinity) under the crucial assumption that the selected system model class and the noise model are correct [1]. However, if these assumptions are not satisfied, PEM methods may yield models that exhibit low accuracy when used to simulate the system. See [10] and [11] for a detailed discussion.

A viable alternative is to formulate the parameter estimation problem as the minimization of the simulation error, i.e., we define $\mathcal{L}$ in (1) as

$$\mathcal{L}(\theta) = \sum_{t=1}^{N} \|y_t(\theta|u) - \tilde{y}_t\|_p \tag{2}$$

where $y_t(\theta|u)$ is the output of the model when the measured input u is provided, and $p = 1, 2$, or ∞. This approach is known as *simulation error minimization* (SEM) and leads to consistent estimates regardless of the measurement noise model; see [12] and [13]. A noteworthy case is the SEM problem for linear systems. In such a scenario, the optimization problem is polynomial, and semidefinite relaxation techniques are adopted to compute the global solution; see, e.g., [14] for theoretical details and [15] for an illustrative example. Conversely, in the general case of nonlinear systems, the SEM problem is a generic differentiable non-convex minimization problem for which we can compute local solutions only.

Several possibilities to minimize (2) are available. The most commonly considered approach is applying gradient-based optimization algorithms. Common choices include first-order gradient descent (GD) algorithms such as stochastic and mini-batch GD, Nesterov's, RMSprop, and Adam algorithms. See, e.g., [16] and [17]. Alternatively to GD, we have second-order methods. These methods present a theoretically guaranteed improvement in the convergence rate compared to first-order algorithms. Nevertheless, they are computationally expensive, and their complexity does not scale well for large problems because they require computing large Hessian matrices. A typical example is Newton's method. To address these limitations, methods utilizing Hessian approximation have emerged. Among them, we mention Gauss-Newton, Levenberg-Marquardt (LM) [18], and Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithms [19], [20], and [21]. We refer to these methods as quasi-second order.

When using gradient-based optimization algorithms, the problem of evaluating the gradients $\nabla_\theta y_t(\theta)$ emerges. The problem's dynamic structure generates a dynamic relation between the variables $\nabla_\theta y_t(\theta)$. The technique known as dynamic backpropagation, proposed in [22], leverages this dependence to evaluate the gradients as the output of a dynamical system called the sensitivity model.
A more popular alternative is the backpropagation through time (BPTT) algorithm; see, e.g., [23] and [24]. In BPTT, the dynamic dependencies are utilized implicitly to compute partial derivatives recursively. Specifically, BPTT relies on unrolling, i.e., expanding the recurrent model over time into a series of feedforward models. In this framework, the standard choice for the computation of the derivatives is using automatic differentiation (see, e.g., [25] for a recent survey).

Solving SEM problems using gradient-based algorithms is generally associated with slow convergence, numerical convergence to non-optimal solutions, or even instability. The root of these issues lies in phenomena known as vanishing and exploding gradient issues; see, e.g., [26] and [27]. These issues arise independently of the method used to compute the gradient. Indeed, the dependence of $\nabla_\theta y_t$ from $\nabla_\theta y_{t-1}$, for all $t = n+1, \dots, N$, implicitly defines a dynamical system whose outputs are the required gradients. If such a system is a contraction, $\nabla_\theta y_t$ becomes vanishingly small for large t. On the other hand, if it is unstable, $\nabla_\theta y_t$ diverges.

Exploding gradient issues may be handled effectively by techniques such as gradient clipping [28]. On the other hand, effectively and efficiently learning arbitrary nonlinear systems without encountering vanishing gradient issues is an open and challenging problem. In particular, since the vanishing

gradient leads to slow convergence and inaccurate estimation, alternative methods avoiding the vanishing gradient issue are required to improve the time and energy needed to identify an accurate model, thus increasing the SI approach's scope of applicability.

A relevant sub-class of SI problems that has garnered recent interest is related to recurrent neural networks (RNNs); see, e.g., [29] and [17]. Known for their capability to approximate any dynamical system, RNNs are now widely employed to perform nonlinear black-box identification [30, 31]. Early works on RNN models led to the formulation of the Elman RNNs and neural state-space models that are classically identified using BPTT. In this context, the classical approach to attenuate the vanishing gradient problem is to include a direct gradient propagation path in the network definition. This idea led to the definition of different RNN structures like long-short-term memory (LSTM) and gated recurrent units (GRU) networks. See, e.g., [17, 32] for a detailed discussion. However, such an approach leads to the definition of large networks that are prone to overfitting and less data-efficient than classical RNNs. Moreover, common regularization strategies that mitigate overfitting may fail when applied to such networks [33].

Despite the practical advantages of LSTM and GRU against classical RNNs, the main idea behind how they attenuate the vanishing gradient issue is based on modifying the system's model; therefore, we cannot directly employ this approach to learn general dynamical models, e.g., gray-box and physics-informed models. This observation motivates the need for alternative approaches to tackle the vanishing gradient problem. Recent contributions in this direction explore defining an identification algorithm that does not require the computation of the gradients $\nabla_\theta y_t$. This approach can avoid vanishing and exploding gradient problems instead of attenuating them. Recent works along this direction include:

- application of meta-heuristic global optimization, e.g., particle swarm optimization or genetic algorithms. See, e.g., [34] and [35]. Although these methods completely avoid any gradient computation, they are generally very slow when estimating many parameters. Moreover, they lack theoretical convergence guarantees.
- [36], where the authors propose considering a constrained optimization formulation of the problem and applying approximated sequential quadratic programming (SQP) to solve it. This approach effectively reduces the number of iterations required for convergence while the cost of each iteration increases.

In this thesis, we consider SEM identification of nonlinear input-output models. Specifically, we formulate the identification problem by defining a constrained optimization problem in which optimization variables represent model parameters and noise-free outputs, while constraints define the relation between them; i.e., we consider

$$\begin{aligned} &\arg\min \theta \in \mathbb{R}^{n_\theta}, Y \in \mathbb{R}^{pN} \mathcal{L}(\theta, y) \\ &\quad \text{s.t.} \\ &\quad h_{t-n}(\theta, y) = -y_t + \mathcal{M}(\theta, y|u) = 0, \\ &\qquad t = n+1, \dots, N, \end{aligned} \tag{3}$$

where $Y = [y_1^\top, \dots, y_N^\top]^\top \in \mathbb{R}^{pN}$, $\mathcal{M} : \mathbb{R}^{n_\theta} \times \mathbb{R}^{pN} \to \mathbb{R}^p$ represents the model of the system and $\mathcal{L} : \mathbb{R}^{n_\theta} \times \mathbb{R}^{pN} \to \mathbb{R}$ accounts for the simulation error. A similar mathematical formulation is considered in [36], where the authors develop an inexact SQP method for training neural state-space models. Moreover, constrained optimization problems of this kind are also formulated in the context of set-membership identification, in which we typically assume that noise is bounded and incorporate additional inequality constraints accordingly. See, e.g., [37, 14].

Let us denote $x = [\theta^\top, y_1^\top, \dots, y_N^\top]^\top \in \mathbb{R}^{n_\theta + pN}$ the optimization variables of Problem (3). This problem is characterized by the first-order optimality conditions

$$\begin{aligned} &\nabla_x \mathcal{L}(x) + \sum_{t=1}^{N-n} \lambda_t \nabla_x h_t(x) = 0 \\ &h_t(x) = 0, \qquad t = 1, \dots, N-n, \end{aligned} \tag{4}$$

which are a set of necessary conditions for the local optimality of x.
Notice that this condition only requires computing gradients of simple functions that do not recursively

depend on expressions evaluated at previous time instants. Consequently, the numerical difficulties related to the vanishing gradient are absent if we use this formulation. We highlight that we can avoid vanishing gradient issues by solving a constrained optimization problem. Nevertheless, this task is challenging from a computational viewpoint.

Standard approaches to constrained optimization include interior-point methods, sequential quadratic programming, and Lagrangian methods; see [38] and [39]. All these approaches require solving large systems of linear equations, which may become computationally and memory-prohibitive when the number of data (and consequently, constraints) increases.

This consideration motivates the need for an efficient equality-constrained optimization algorithm. For this purpose, this thesis aims to develop novel, first-order, memory-efficient algorithms for constrained optimization. Specifically, we propose an original approach based on feedback control theory, which we call the *controlled multipliers optimization* (CMO) framework.

CMO involves reformulating an assigned constrained optimization problem into an equivalent stabilization and output regulation control problem. We define a fictitious plant where first-order optimality conditions define the state equations, the constraints define the output, and the Lagrange multipliers are interpreted as the control input. We demonstrate that by finding appropriate control laws, we can define a feedback control system that drives the plant's state trajectories to converge toward the optimization problem solution.

We focus on equality-constrained optimization problems and present two solutions: one based on proportional-integral (PI) control [40], and the other based on feedback linearization (FL) [41]. We refer to the resulting algorithms as PI-CMO and FL-CMO, respectively. Our theoretical analysis demonstrates that both algorithms achieve global exponential convergence to the global optimal solution to the optimization problem when the objective function is convex and the constraints are linear. Additionally, we conduct a local analysis of FL-CMO to establish the asymptotic stability of isolated minima for non-convex problems.

Next, we address optimization problems that are subject to linear inequality constraints. These problems often arise in set-membership identification, particularly when the noise is assumed to be bounded in magnitude. We propose a solution based on the CMO framework, considering a plant defined using the augmented Lagrangian and a modified PI controller; we call this algorithm the modified PI-CMO (M-PI-CMO). We show that this algorithm is globally exponentially convergent for convex optimization problems.

Finally, we compare our approaches with similar ones in the literature. Specifically, we compare PI-CMO and M-PI-CMO against primal-dual gradient dynamics (PDGD). Both theoretical and simulation results indicate that PI-CMO and M-PI-CMO converge faster than PDGD. Furthermore, FL-CMO is compared to null-space gradient dynamics, demonstrating that FL-CMO generalizes the latter to a novel class of algorithms.

In the second part of the thesis, we apply the proposed FL-CMO optimization algorithm to the problem of identifying nonlinear input-output (NIO) models. Specifically, we consider a rather general class of linear or nonlinear models in the form

$$y_t = \mathcal{M}(y_{t-1}, \dots, y_{t-n}, u_t, \dots, u_{t-n}|\theta), \tag{5}$$

where $u_t \in \mathbb{R}^q$ is the input, $y_t \in \mathbb{R}^p$ is the output, and

$$\mathcal{M} : \underbrace{\mathbb{R}^p \times \cdots \times \mathbb{R}^p}_{n \text{ times}} \times \underbrace{\mathbb{R}^q \times \cdots \times \mathbb{R}^q}_{n+1 \text{ times}} \to \mathbb{R}^p \tag{6}$$

is any differentiable function. The considered model class can account for a large array of models, including linear, block-structured, gray-box, physics-informed, and neural network models.

We formulate the identification problem as a constrained optimization of the kind (3) and demonstrate that the structure of the problem is compatible with the assumptions required to apply FL-CMO. Next, we prove that it converges to a stationary point of the original unconstrained formulation (1). To cope with the main computational bottleneck of the algorithm, we leverage the sparsity of the constraints' Jacobian to propose a Q-less QR factorization procedure that allows us to reduce the computational complexity from cubic to quadratic in the number of identification data samples. Moreover, we optimize the amount of required iterations by using an adaptive step size to integrate the original, continuous-time formulation of FL-CMO.

In addition to introducing the algorithm and performing its theoretical analysis, we demonstrate the effectiveness of the proposed approach on several SI problems. We consider four key problems: established black-box SI benchmarks and a realistic gray-box identification problem. Concerning the black-box problems, we consider the *nonlinear neural output error* (NNOE) model, which is a particular kind of RNN in input-output form. The results indicate that, on the one hand, our approach significantly outperforms standard methods for training NNOE models. On the other hand, a properly trained NNOE model can provide superior performances compared to prevalent RNN models, including Elman RNN, LSTM, and GRU. We explain the latter by noting that NIO models are intrinsically more data-efficient than their state-space counterparts. Concerning gray-box identification, we show that the proposed method significantly improves the parameter estimates compared to standard gradient approaches.

Finally, as a side project, we also devote our attention to the identification of linear-time-invariant systems. We formulate the SEM problem for this class of models as constrained optimization, and we notice that the formulated problem is described by polynomials. This observation motivates the need to develop novel and fast algorithms for polynomial optimization. We address this problem by proposing a solution based on the alternating direction method of multipliers (ADMM) algorithm, which we call ADMM4POP. We theoretically analyze the algorithm's convergence and validate the effectiveness of the proposed approach through numerical examples.

3 Outline of the thesis

This thesis is organized into three parts.

3.1 Part I: Background

Part I provides a background of the required preliminary results needed in the subsequent chapters.

In Chapter 2, we deal with the *nonlinear SI problem*. First, we introduce the general SI problem, review the relationship between state-space and input-output models, and discuss the differences between commonly considered loss functions used to define the problem. Then, we provide an overview of SI methods, focusing on linear, gray-box, block-oriented, and neural network models.

In Chapter 3, we recall definitions and standard results on *nonlinear systems* stability and control. Specifically, we first deal with notions of stability and review Lyapunov's stability criteria; then, we review feedback linearization controller design for nonlinear systems.

Chapter 4 reviews basic notions and fundamental results related to *optimization problems*. We consider constrained and unconstrained optimization problems and review results on optimality conditions. Next, we review some optimization algorithms with a particular emphasis on gradient and Lagrangian methods. Finally, we review some results related to dynamical systems approaches to the analysis of optimization algorithms' convergence.

3.2 Part II: Constrained optimization through control

Part II is the core of the thesis. It introduces the proposed *controlled multipliers optimization* (CMO) approach to design optimization algorithms through control theory.

In Chapter 5, we introduce the main proposed *framework* by defining the fictitious plant to be controlled and proving two fundamental lemmas: one for equality-constrained problems and another for inequality-constrained ones.

Chapter 6 proposes an optimization algorithm based on *PI control* (PI-CMO) for equality-constrained optimization. We prove its convergence for strongly convex optimization problems and compare it to primal-dual gradient dynamics (PDGD).

In Chapter 7, we consider optimization problems with *linear inequality* constraints and propose a solution based on a modification of the PI control law (modified PI-CMO). We theoretically analyze its convergence for convex problems and compare it with the augmented Lagrangian PDGD (Aug-PDGD).

In Chapter 8, we develop a solution based on *feedback linearization* controller design (FL-CMO). We prove convergence for both convex and non-convex optimization problems and demonstrate the practical effectiveness of the approach through numerical experiments.

This part is partially based on the following papers:

- [42]. V. Cerone, S. M. Fosson, S. Pirrera, D. Regruto, "A new framework for constrained optimization via feedback control of Lagrange multipliers.", submitted to IEEE Transactions on Automatic Control, 2024, available at https://arxiv.org/abs/2403.12738.

- [43]. V. Cerone, S. M. Fosson, S. Pirrera, D. Regruto, "A feedback control approach to convex optimization with inequality constraints," In 63rd IEEE Conference on Decision and Control (CDC), 2024, available at https://arxiv.org/ abs/2409.07168.

3.3 Part III: Constrained optimization for system identification

Part III deals with applying algorithms for constrained optimization to identify dynamical systems.

In Chapter 9, we consider *nonlinear input-output models* and formulate the identification problem as constrained optimization. Then, we propose a learning algorithm defined starting from FL-CMO, as proposed in Chapter 8. We introduce novel strategies to improve the algorithm's computational cost, and we theoretically study the computational complexity of the iterations, proving that it is quadratic in the number of data. Finally, we present numerical results on benchmarks and selected problems concerning both black-box and gray-box SI.

Chapter 10 is devoted to the *linear SI* problem. In particular, we formulate the linear SI problem as a polynomial optimization problem and propose an algorithm based on ADMM to solve it efficiently. We analyze the convergence of the proposed algorithm and provide numerical examples.

This part is partially based on the following papers:

- [44]. V. Cerone, S. M. Fosson, S. Pirrera, D. Regruto, "A constrained optimization approach to system identification of nonlinear input-output models," manuscript in preparation, 2024.

- [15]. V. Cerone, S. M. Fosson, S. Pirrera, D. Regruto, "Alternating direction method of multipliers for polynomial optimization," In European Control Conference (ECC), 2023, available at https://ieeexplore.ieee.org/document/ 10178190.

Chapter 11 draws the conclusions of the thesis.

References

[1] L. Ljung, *System Identification: Theory for the User*. Prentice Hall PTR, 1999.

[2] D. J. Wagg, K. Worden, R. J. Barthorpe, and P. Gardner, "Digital twins: state-of-the-art and future directions for modeling and simulation in engineering dynamics applications," *ASCE-ASME J. Risk Uncertain. Eng. Syst. B: Mech. Eng.*, vol. 6, no. 3, p. 030901, 2020.

[3] M. Milanese and M. Taragna, "H_∞ set membership identification: A survey," *Automatica*, vol. 41, no. 12, pp. 2019–2032, 2005.

[4] J. A. Rossiter, *Model-based predictive control: a practical approach*. CRC press, 2017.

[5] J. M. Hokanson, G. Iaccarino, and A. Doostan, "Simultaneous identification and denoising of dynamical systems," *SIAM J. Sci. Comput.*, vol. 45, no. 4, pp. A1413–A1437, 2023.

[6] A. Karimi, K. V. Heusden, and D. Bonvin, "Non-iterative data-driven controller tuning using the correlation approach," in *Proc. Eur. Control Conf. (ECC)*, pp. 5189–5195, IEEE, 2007.

[7] M. C. Campi and S. M. Savaresi, "Direct nonlinear control design: The virtual reference feedback tuning (VRFT) approach," *IEEE Trans. Autom. Control*, vol. 51, no. 1, pp. 14–27, 2006.

[8] M. Abuabiah, V. Cerone, S. Pirrera, and D. Regruto, "A non-iterative approach to direct data-driven control design of MIMO LTI systems," *IEEE Access*, vol. 11, pp. 121671–121687, 2023.

[9] G. Pillonetto, F. Dinuzzo, T. Chen, G. D. Nicolao, and L. Ljung, "Kernel methods in system identification, machine learning and function estimation: A survey," *Automatica*, vol. 50, no. 3, pp. 657–682, 2014.

[10] Q. Zhang, "Nonlinear system identification with output error model through stabilized simulation," *Proc. Symp. Nonlinear Control Syst. (NOLCOS)*, vol. 37, pp. 501–506, 2004.

[11] L. Piroddi, "Simulation error minimisation methods for narx model identification," *Int. J. Model. Identif. Control*, vol. 3, no. 4, pp. 392–403, 2008.

[12] T. Söderström and P. Stoica, "Some properties of the output error method," *Automatica*, vol. 18, no. 1, pp. 93–99, 1982.

[13] M. Farina and L. Piroddi, "Convergence properties of an iterative prediction approach to nonlinear sem parameter estimation," in *IEEE Conf. Decis. Control (CDC)*, pp. 7226–7231, IEEE, 2010.

[14] V. Cerone, J.-B. Lasserre, D. Piga, and D. Regruto, "A unified framework for solving a general class of conditional and robust set-membership estimation problems," *IEEE Trans. Autom. Control*, vol. 59, no. 11, pp. 2897–2909, 2014.

[15] V. Cerone, S. M. Fosson, S. Pirrera, and D. Regruto, "Alternating direction method of multipliers for polynomial optimization," in *Proc. Eur. Control Conf. (ECC)*, pp. 1–6, 2023.

[16] I. Sutskever, *Training recurrent neural networks*. University of Toronto Toronto, ON, Canada, 2013.

[17] F. M. Salem, *Recurrent Neural Networks*. Springer, 2022.

[18] D. T. Mirikitani and N. Nikolaev, "Recursive bayesian recurrent neural networks for time-series modeling," *IEEE Trans. Neural Netw.*, vol. 21, no. 2, pp. 262–274, 2009.

[19] C.-C. Peng and G. D. Magoulas, "Nonmonotone BFGS-trained recurrent neural networks for temporal sequence processing," *Appl. Math. Comput.*, vol. 217, no. 12, pp. 5421–5441, 2011.

[20] X. Liu, S. Liu, J. Sha, J. Yu, Z. Xu, X. Chen, and H. Meng, "Limited-memory bfgs optimization of recurrent neural network language models for speech recognition," in *Proc IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, pp. 6114–6118, IEEE, 2018.

[21] A. Bemporad, "Linear and nonlinear system identification under ℓ_1-and group-lasso regularization via L-BFGS-B," *arXiv preprint arXiv:2403.03827*, 2024.

[22] K. S. Narendra and K. Parthasarathy, "Gradient methods for the optimization of dynamical systems containing neural networks," *IEEE Trans. Neural Netw.*, vol. 2, no. 2, pp. 252–262, 1991.

[23] R. J. Williams, "A learning algorithm for continually running fully recurrent neural netwokrs," *Neural Comput.*, vol. 1, pp. 256–263, 1989.

[24] P. J. Werbos, "Backpropagation through time: what it does and how to do it," *Proc. IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.

[25] A. G. Baydin, B. A. Pearlmutter, A. A. Radul, and J. M. Siskind, "Automatic differentiation in machine learning: a survey," *J. Marchine Learn. Res.*, vol. 18, pp. 1–43, 2018.

[26] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, 1994.

[27] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, pp. 1310–1318, Pmlr, 2013.

[28] A. Ramaswamy, "Gradient clipping in deep learning: A dynamical systems perspective," in *Int. Conf. Pattern Recognit. Appl. Methods (ICPRAM)*, vol. 1, pp. 107–114, 2023.

[29] L. Medsker and L. C. Jain, *Recurrent neural networks: design and applications*. CRC press, 1999.

[30] L. Ljung, C. Andersson, K. Tiels, and T. B. Schön, "Deep learning and system identification," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 1175–1181, 2020.

[31] K. Benidis, S. S. Rangapuram, V. Flunkert, Y. Wang, D. Maddix, C. Turkmen, J. Gasthaus, M. Bohlke-Schneider, D. Salinas, L. Stella, *et al.*, "Deep learning for time series forecasting: Tutorial and literature survey," *ACM Comput. Surv.*, vol. 55, no. 6, pp. 1–36, 2022.

[32] A. Shewalkar, D. Nyavanandi, and S. A. Ludwig, "Performance evaluation of deep neural networks applied to speech recognition: RNN, LSTM and GRU," *J. Artif. Intell. Soft Comput. Res.*, vol. 9, no. 4, pp. 235–245, 2019.

[33] S. Merity, N. S. Keskar, and R. Socher, "Regularizing and optimizing LSTM language models," in *Proc. Int. Conf. Learn. Represent. (ICRL)*, 2018.

[34] A. Blanco, M. Delgado, and M. C. Pegalajar, "A real-coded genetic algorithm for training recurrent neural networks," *Neural Netw.*, vol. 14, no. 1, pp. 93–105, 2001.

[35] E. Bas, E. Egrioglu, and E. Kolemen, "Training simple recurrent deep artificial neural network for forecasting using particle swarm optimization," *Granular Computing*, vol. 7, no. 2, pp. 411–420, 2022.

[36] A. D. Adeoye and A. Bemporad, "An inexact sequential quadratic programming method for learning and control of recurrent neural networks," *IEEE Trans. Neural Netw. and Learning Systems*, 2024.

[37] M. Milanese, J. Norton, H. Piet-Lahanier, and É. Walter, *Bounding Approaches to System Identification*. Springer US, 2013.

[38] D. P. Bertsekas, *Nonlinear programming*. Athena Scientific, 3rd ed., 2016.

[39] J. Nocedal and S. Wright, *Numerical optimization*. Springer, 2006.

[40] A. O'dwyer, *Handbook of PI and PID controller tuning rules*. World Scientific, 2009.

[41] A. Isidori, *Nonlinear Control Systems*. Springer London, 1995.

[42] V. Cerone, S. M. Fosson, S. Pirrera, and D. Regruto, "A new framework for constrained optimization via feedback control of lagrange multipliers," *arXiv preprint arXiv:2403.12738*, 2024.

[43] V. Cerone, S. M. Fosson, S. Pirrera, and D. Regruto, "A feedback control approach to convex optimization with inequality constraints," in *Proc. IEEE Conf. Decis. Control (CDC)*, IEEE, 2024.

[44] V. Cerone, S. M. Fosson, S. Pirrera, and D. Regruto, "A constrained optimization approach to system identification of nonlinear input-output models," *Manuscript in preparation*, 2024.