2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

A collaborative multi-agent nonlinear system identification algorithm with spectral regularization

Simone Smeraldo[1,2], Federico Bianchi[3], Axel Busboom[2], Maria Prandini[1]

***Abstract*— We address nonlinear system identification in a multi-agent setting, where each agent collects input-output data from a different instance of the same process, possibly in a different operating condition. In particular, we introduce a novel scheme for nonlinear auto-regressive with exogenous input (NARX) model identification where agents make a tentative estimate of their individual parameters, based on local information only, while a cloud-based application further manipulates these estimates so as to disclose the common model structure and adjust the values of the individual parameters around some reference parameter vector. The proposed scheme is inspired by the spectral regularization framework recently introduced in multi-task feature learning and is shown to be competitive against state-of-the art cloud-based algorithms addressing the same problem but under more restrictive assumptions.**

I. INTRODUCTION

The increasing use of connected devices and their boost in computational power is progressively enabling the proliferation of cloud-based systems [1]. Data collected and partially processed at the edge level by local units (the agents) can be communicated to a central unit at the cloud level for further more computationally intensive operations. If the devices are similar, data can be jointly exploited to enhance the performance at the single device level, in applications including estimation, monitoring, prediction, diagnostics.

In this paper, we address the identification of similar yet not identical models for multiple devices that are operating in possibly different conditions and/or are configured differently, such as, for instance, in the case of a fleet of industrial assets or of vehicles, and in a micro-grid aggregating multiple prosumers.

In particular, we focus on the identification of nonlinear discrete time dynamical models, a challenging problem which has been extensively studied in the last few decades [2]. A frequently adopted discrete time system representation is the Nonlinear Auto-Regressive eXogenous input (NARX) model [3], which consists in a nonlinear functional expansion of lagged inputs and outputs, often expressed in a polynomial form, with a *linear-in-the-parameters* structure that is particularly convenient for parameter estimation purposes. On the other hand, the number of terms in the expansion grows rapidly with the model order and nonlinearity degree, which motivates the need for the selection of the appropriate terms to be included. Model structure selection (MSS) is a combinatorial problem that is hard to handle through an exhaustive search or statistical indices like the Bayesian Information Criterion (BIC), [4], employed in the linear case. This has motivated the development of heuristic methods for the identification of a parsimonious model, such as [5], [6].

The problem of joint MSS and parameter estimation becomes even more complex in the framework considered in this paper. A multi-agent cooperative identification algorithm is adopted in [7], where a cloud-aided strategy based on the Alternating Direction Method of Multipliers (ADMM) is used for the estimation of both global and local models, where surrogates of data only are transmitted to a central cloud to leverage the similarities between agents. The method assumes that the model structure is known and that the parameters belong to some compact sets. A distributed scheme for joint MSS and parameter estimation of a global polynomial NARX model is instead proposed in [8], assuming identical parameters for all the local models. Accurate identification results are achieved even when only a few agents have informative data.

We present in this paper a novel decentralized multi-agent system identification and structure selection algorithm. Inspired by some results in the multi-task learning literature [9], we include in the identification cost function a spectral regularization term [10], which favors the model sharing some common structure, while local data reveal whether the local parameters vectors are identical or not. Our approach to decentralized nonlinear dynamical system identification and structure

[1]Politecnico di Milano, Italy maria.prandini@polimi.it
[2]Munich University of Applied Sciences, Germany firstname.lastname@hm.edu
[3]Ricerca sul Sistema Energetico (RSE) S.p.A., Milano, Italy federico.bianchi@rse-web.it

This work was supported in part by the Bavarian Ministry of Economic Affairs, Regional Development and Energy (StMWi) under grant DIK0237/02.

selection rests on the optimization scheme proposed in the multi-task feature learning algorithm [11]. The resulting constrained optimization problem is convex and computationally appealing, since it can be solved efficiently by a block coordinate descent algorithm exploiting the closed form of the block-minimizations throughout the iterations, as in [11]. Convergence of the proposed algorithm can be proven based on the results in [12]. The proposed approach is able to estimate both shared and local parameters like the scheme in [7] and shares the same communication structure. It also achieves the improved identification performance and structure selection reliability of the scheme in [8], without the restrictive assumptions of [7] and [8].

The remainder of the paper is structured as follows. Section II introduces the NARX model identification problem. The proposed collaborative identification algorithm is presented in Section III and its performance is discussed through numerical simulations in Section IV. Finally, in Section V some conclusions are drawn.

II. PROBLEM STATEMENT

Consider N systems with scalar input u_n and output y_n, $n = 1, \dots, N$, and the same NARX structure:

$$y_n(t) = g(x_n(t); w_n) + e_n(t), \tag{1}$$

where vector $x_n(t) = [y_n(t-1) \dots y_n(t-n_y)\ u_n(t-1) \dots u_n(t-n_u)]^T$ collects lagged input and output (n_y and n_u being suitable maximum lags), $e_n(t)$ is a scalar zero-mean additive white noise, and $g(\cdot; w_n)$ is a nonlinear function, common to all agents, parametrized via a vector $w_n \in \mathbb{R}^d$ of local coefficients. If we express the nonlinear mapping $g(\cdot; w_n)$ as a polynomial functional expansion, then, system (1) takes the form:

$$y_n(t) = \phi_n(t)^T w_n + e_n(t),\ n = 1, \dots, N, \tag{2}$$

where $\phi_n(t) = \phi(x_n(t))$ is the regressor vector whose elements $\phi_n^j(t)$, $j = 1, \dots, d$, are monomials of the lagged input and output in $x_n(t)$ up to some degree n_d.

Suppose that N input-output data sets $\mathcal{D}_n = \{x_n(t), y_n(t)\}_{t=1}^{T_n}$ are collected separately, possibly in different experimental set-ups. Let σ_n^2, $n = 1, \dots, N$, denote the corresponding output process variances. The identification of the local parameter vectors w_n in (2) is formulated as the following constrained optimization problem:

$$\begin{aligned} \min_{\{w_n\}, w_0, D}\ & \frac{1}{N} \sum_{n=1}^{N} \Big[L_n(w_n) \\ & + \gamma (w_n - w_0)^T D^\dagger (w_n - w_0) \Big] + \beta \|w_0\|_2^2 \end{aligned} \tag{3}$$

$$\begin{aligned} \text{subject to: } & D \in \mathcal{S}_+^d \\ & \text{tr}(D) \le 1 \\ & \text{range}(W - W_0) \subseteq \text{range}(D) \end{aligned}$$

where $W \in \mathbb{R}^{d \times N}$ is the matrix whose columns are the agents parameters vectors w_n, $n = 1, \dots, N$, $W_0 \in \mathbb{R}^{d \times N}$ has all columns identical to $w_0 \in \mathbb{R}^d$, $\mathcal{S}_+^d$ denotes the set of $d \times d$ real symmetric positive semidefinite matrices, $D^\dagger$ is the pseudoinverse of D, and $L_n : \mathbb{R}^d \to \mathbb{R}$ defined as

$$L_n(w) = \frac{1}{\sigma_n^2 T_n} \left(\sum_{t=1}^{T_n} \left(y_n(t) - \phi_n(t)^T w \right)^2 + \alpha_n \|w\|_2^2 \right) \tag{4}$$

is a standard least squares cost with L_2 regularization and weight $\alpha_n \in \mathbb{R}^+$, and accounts for the accuracy of the identified NARX local model with parameter vector w on the data set $\mathcal{D}_n$.

In problem (3), $w_0 \in \mathbb{R}^d$ plays the role of a reference parameter vector shared across the agents, whose L_2-norm is penalized through $\beta \in \mathbb{R}^+$ to avoid overfitting in the case of unknown model structure. The deviation of each local parameter vector from w_0 enters the cost function through a quadratic contribution weighted with the matrix $D^\dagger$, to be jointly optimized with w_0 and $\{w_n\}_{n=1}^N$. Coefficient $\gamma \in \mathbb{R}^+$ weights this term. If we express D through its eigenvalue decomposition, i.e., $D = U \Lambda U^T$, with $U \in \mathbb{R}^{d \times d}$ orthogonal and Λ diagonal containing the eigenvalues $\lambda_i \ge 0$, $i = 1, \dots d$, of D, then,

$$D^\dagger = U \text{diag} \left(\lambda_1^\dagger \dots \lambda_d^\dagger \right) U^T \tag{5}$$

where $\lambda^\dagger = \frac{1}{\lambda}$ for $\lambda \neq 0$ and $\lambda^\dagger = 0$ otherwise. Term

$$\begin{aligned} & \sum_{n=1}^{N} (w_n - w_0)^T D^\dagger (w_n - w_0) = \\ & \sum_{n=1}^{N} (U^T (w_n - w_0))^T \text{diag} \left(\lambda_1^\dagger \dots \lambda_d^\dagger \right) U^T (w_n - w_0) \end{aligned} \tag{6}$$

is a special instance of the spectral regularizers for multi-task structure learning discussed in [10] and it accounts for the dispersion of the parameter vectors w_n around w_0. More precisely, $w_n - w_0$ is projected onto the axes of the orthogonal coordinate system defined by U and the resulting features are weighted with the corresponding $\lambda_i^\dagger$'s. To minimize the contribution of $(w_n - w_0)^T D^\dagger (w_n - w_0)$ to the sum in (6), D should be such that λ_i is large if feature i has a large dispersion and small in the opposite case. Since in (3) the sum of the λ_i's is forced to be smaller than 1, then, they cannot

all be large. This constraint together with the condition on the range is encouraging the discovery of common features. A term of the form (6) is indeed employed in the Multi Task Feature Learning algorithm [11] to promote a sparse common (latent) feature representation shared by multiple tasks.

Model structure selection

In order to perform MSS, we adopt the Student's t-test as in [6] to establish the statistical relevance of each regressor based on the optimal estimates w_n resulting from (3). In particular, denoting by $t_{\delta,N-d}$ the $100(1-\delta)$ percentile of the Student's t distribution with $N-d$ degrees of freedom, then the $100(1-\delta)\%$ confidence interval for each w_n^j, namely the jth element in w_n, is given by:

$$[w_n^j - \hat{\sigma}_j t_{\delta,N-d} \ , \ w_n^j + \hat{\sigma}_j t_{\delta,N-d}] \tag{7}$$

$\hat{\sigma}_j$ being the variance of the estimated parameters, that can be estimated as: $\hat{\sigma}_j^2 \approx \hat{\sigma}_e^2 P_n^{jj}$, where $\hat{\sigma}_e^2$ is the estimated noise variance, obtained by scaling the mean squared residual by a factor $N/(N-d)$, and P_n^{jj} is the jth diagonal element of

$$P = \left(\sum_{t=1}^{T_n} \phi_n(t) \phi_n^T(t) \right)^{-1}.$$

If the interval (7) does not contain zero, w_n^j is not zero with confidence of $100(1-\delta)\%$. Otherwise, w_n^j is considered to be statistically irrelevant by agent n. The corresponding monomial ϕ_n^j is removed from the regressor vector only if all agents consider the jth component w_n^j of their local parameter vector w_n statistical irrelevant.

III. PROPOSED ALGORITHM

In this section we introduce a decentralized algorithm to solve problem (3) resting on block-coordinate descent. We start by noticing that problem (3) is convex as shown in [11], where a similar cost function is considered. For the minimizer to be uniquely defined and the block-coordinate descent algorithm to converge, a cost perturbation is suggested in [11]. Accordingly, we modify (3) as follows:

$$\begin{aligned} \min_{\{w_n\},w_0,D} \ & \frac{1}{N} \sum_{n=1}^{N} L_n(w_n) + \beta \|w_0\|_2^2 \\ & + \frac{\gamma}{N} \operatorname{tr}\left(D^{-1} \left((W - W_0)(W - W_0)^T + \varepsilon I_d \right) \right) \\ \text{subject to:} \ & D \in \mathcal{S}_{++}^d \\ & \operatorname{tr}(D) \leq 1 \end{aligned} \tag{8}$$

where $\mathcal{S}_{++}^d$ denotes the set of $d \times d$ real symmetric positive definite matrices and the regularizer in (3) has been equivalently written in trace form, with the inclusion of εI_d, where I_d denotes the $d \times d$ identity matrix. Indeed, the perturbation keeps D non singular, hence the use of D^{-1} and the removal of the range constraint.

In the first step of our algorithm, each agent computes a local estimate w_n^l of its parameter vector w_n by minimizing $L_n(w_n)$ in (4), thus getting

$$w_n^l = V_n \frac{1}{\sigma_n^2 T_n} \Phi_n Y_n, \tag{9}$$

where $\Phi_n = [\phi_n(1) \dots \phi_n(T_n)]$, $Y_n = [y_n(1) \dots y_n(T_n)]^T$, and

$$V_n = \left(\frac{1}{\sigma_n^2 T_n} \Phi_n \Phi_n^T + \frac{1}{\sigma_n^2 T_n} \alpha_n I_d \right)^{-1} \tag{10}$$

can be computed locally by each agent. The agents transmit then w_n^l to the central cloud-based unit, which, based on these local estimates, solves the convex problem (8) via block-coordinate descent, thus obtaining w_n, $n = 1, \dots, N$, which are then sent back to the agents to perform local t-tests using private data only. The test outcome is codified in a binary vector $b_i \in \{0,1\}^d$, whose j-th element is 0 if w_n^j is considered statistically irrelevant by agent n, and 1 otherwise. Finally, these binary vectors are sent to the cloud, where the common model structure is coded by the binary vector

$$b = b_1 \vee b_2 \vee \dots \vee b_N, \tag{11}$$

which has zero components only for those elements of the parameter vector that are considered statistically irrelevant by all the agents.

Algorithm 1 shows a pseudo-code description of the proposed procedure. Information exchange in parameters estimation has to be carried out only once, since the iterations are all performed at the cloud level. This communication scheme is more efficient than the one proposed in [13], whose algorithm entails multiple exchanges between agents and central unit for each cycle.

We next derive the equations referenced in Algorithm 1 for computing all relevant quantities while running the sequential steps of the block-coordinate descent.

Block-coordinate descent sequential steps

We start deriving the expression of w_0 minimizing the cost function in (8) for given w_n, $n = 1, \dots, N$, and D, which will turn out to be a function only of D and the local estimates w_n^l. To this purpose, we first compute w_n as a function of D and w_0 by setting equal to zero the derivative of the cost in (8) with respect to w_n, thus

Algorithm 1 Collaborative multi-agent identification

Require: $\{\Phi_n,\ Y_n\}_{n=1}^N$, γ, α_n, β, ε, $D_{ini} = I_d \cdot \frac{1}{d}$, tol_W

1. $D \leftarrow D_{ini}$
2. **Each Agent:**
   2.1. Compute V_n as in (10) and w_n^l as in (9)
   2.2. Transmit V_n, w_n^l to the central unit
3. **Central Unit:**
   3.1. **While** $\|W - W_{prev}\| > \|W\| * tol_W$ **do**
      3.1.1. Update w_0 as in (17)
      3.1.2. Update w_n as in (15) for $n = 1, \dots, N$
      3.1.3. Update D as in (19)
   3.2. Transmit w_n to agent n, for $n = 1, \dots, N$
4. **Each Agent:**
   4.1. Compute b_n
   4.2. Transmit b_n to the central unit
5. **Central Unit:**
   5.1. Compute b as in (11)
   5.2. Transmit b to the agents.

getting:

$$w_n = \left(V_n^{-1} + \gamma D^{-1}\right)^{-1} \frac{1}{\sigma_n^2 T_n} \Phi_n Y_n + \left(V_n^{-1} + \gamma D^{-1}\right)^{-1} \gamma D^{-1} w_0. \tag{12}$$

By the matrix inversion lemma we can express the inverse of $V_n^{-1} + \gamma D^{-1}$ appearing in (12) as

$$V_n - V_n \gamma D^{-1} \left(I_d + V_n \gamma D^{-1}\right)^{-1} V_n,$$

which is equal to $\left(I_d + V_n \gamma D^{-1}\right)^{-1} V_n$. We can then rewrite (12) as

$$w_n = \left(I_d + V_n \gamma D^{-1}\right)^{-1} \left(w_n^l + V_n \gamma D^{-1} w_0\right), \tag{13}$$

where w_n^l is given in (9). If we now define

$$\hat{w}_n = \left(I_d + V_n \gamma D^{-1}\right)^{-1} w_n^l \qquad Z_n = \left(I_d + V_n \gamma D^{-1}\right)^{-1} \tag{14}$$

we can rewrite (13) as

$$w_n = \hat{w}_n + Z_n V_n \gamma D^{-1} w_0. \tag{15}$$

Taking the derivative of (8) with respect to w_0 and setting it equal to zero, we get:

$$w_0 = \left(\beta I_d + \gamma D^{-1}\right)^{-1} \frac{\gamma}{N} D^{-1} \sum_{n=1}^N w_n$$

which using (15) becomes:

$$w_0 = \left(\beta I_d + \gamma D^{-1}\right)^{-1} \frac{\gamma}{N} D^{-1} \sum_{n=1}^N \left(\hat{w}_n + Z_n V_n \gamma D^{-1} w_0\right).$$

By defining

$$\hat{w}_0 = \left(\beta I_d + \gamma D^{-1}\right)^{-1} \frac{\gamma}{N} D^{-1} \sum_{n=1}^N \hat{w}_n, \tag{16}$$

$$Z_0 = \left(\beta I_d + \gamma D^{-1}\right)^{-1} \frac{\gamma}{N} D^{-1} \sum_{n=1}^N Z_n V_n$$

we finally obtain

$$w_0 = \left(I_d - \gamma Z_0 D^{-1}\right)^{-1} \hat{w}_0, \tag{17}$$

which depends on D and w_n^l, $n = 1, \dots, N$ through (16) and (14).

In the second step of the block-coordinate descent method, we compute w_n, $n = 1, \dots, N$, that minimizes the cost function in (8). This is straightforward given the previous calculations: we just need to plug in the value of w_0 obtained from (17) into (15).

In the third block-coordinate descent step, w_n, $n = 1, \dots, N$, and w_0 are fixed, and we just need to minimize with respect to D the regularization cost

$$\min_D \operatorname{tr}\left(D^{-1}\left((W - W_0)(W - W_0)^T + \varepsilon I_d\right)\right) \quad \text{subject to: } \{\operatorname{tr}(D) \leq 1,\ D \in \mathcal{S}_{++}^d\} \tag{18}$$

As shown in [10], this problem admits the optimal solution

$$D(W) = \frac{((W - W_0)(W - W_0)^T + \varepsilon I)^{\frac{1}{2}}}{\operatorname{trace}[((W - W_0)(W - W_0)^T + \varepsilon I)^{\frac{1}{2}}]}, \tag{19}$$

Interestingly, in [10] problems of the form of (18) are shown to reduce to the singular value decomposition (SVD) of $(W - W_0)$, and matrix D, together with its inverse D^{-1} used in the subsequent iteration of the block coordinate descent, can be simply computed through vector operations on the obtained SVD.

IV. NUMERICAL EXAMPLES

We consider three different scenarios. In the first one, agents have to jointly identify the structure and parameters of different instances of the same process. In the second one, they need to estimate the parameters of the same model with a known structure but from data collected in different experimental setups. In the last scenario, accuracy and scalability of the proposed algorithm are examined by considering a given amount of data and dividing it equally among a growing number of agents. The algorithm for parameter estimation proposed in [7] (hereafter referred to as ADMM-RLS) has been applied to the last two scenarios for comparative purposes under the assumption that local model parameters, besides

being identical among the agents, belong to a known compact set.

Scenario 1

We consider data generated by $N = 4$ NARX systems of the form (2) with regressor vector

$$\phi_n(t) = [y_n(t-1) \quad u_n(t-1) \quad u_n(t-1)^2 \quad u_n(t-1)^3]^\top$$

and w_n reported in Table I, which were extracted from a Gaussian distribution with mean $[0.8, 0.4, 0.4, 0.4]^\top$ and covariance matrix $0.1I_4$, where I_4 denotes the identity matrix of size 4. Each agent collects a dataset of length 5000 obtained with $u_n(t) \sim WGN(0, 0.333)$ and $e_n(t) \sim WGN(0, 0.1)$, $WGN(\mu, \sigma^2)$ denoting a White Gaussian Noise with mean μ and variance σ^2.

Model selection is performed over a candidate regressor pool including all monomials up to lags $n_y = n_u = 3$ and maximum degree $n_d = 3$, for a total of 84 terms.

A Monte Carlo analysis has been carried out by running Algorithm 1 on 100 different realizations of $u_n(t)$ and $e_n(t)$, using the parameters in Table III. A non-null β is employed to regularize the value of w_0 over spurious regressors. Table I reports the parameters estimation and structure selection results. The numerical experiment confirms that, when the actual system structure is included in the model pool, then our algorithm is able to correctly identify it together with the local parameters.

Scenario 2

We consider the same NARX system

$$y_n(t) = 0.5y_n(t-1) + 0.8u_n(t-1) + 0.1u_n(t-1)^2 + e_n(t), \tag{20}$$

for $N = 4$ agents, and assume that each one collects a dataset of length 5000 but in a different experimental setting. In particular, $u_n(t) \sim WGN(0, 0.0001)$, $n = 1, 2, 3$ and $u_4(t) \sim WGN(0, 1)$, while $e_n(t) \sim WGN(0, 0.0001)$, $n = 1, 2, 3, 4$. The first 3 agents are then referred to as "non-informative", since the nonlinear dynamics in system (20) is not excited enough to identify the model structure based on each dataset separately. We perform a Monte Carlo analysis by extracting 100 parameters instances of system (20) from a normal distribution with mean $\overline{w} = [0.5,\ 0.8,\ 0.1]$ and covariance matrix $C = \text{diag}([0.4, 5, 5])$. Blindness of the data collected with respect to the presence of the nonlinear regressor is verified for every instance by the Orthogonal Forward Regression procedure in [5]. The ADMM-RLS algorithm is run with the parameters in Table II and with known uncertainties on the parameters values of $\pm\{1, 10, 15\}\%$. Algorithm 1 uses the parameters in Table III.

Figure 1 shows the distribution of the relative error on the first entry global parameters estimate (w_0 in our algorithm) resulting from the Monte Carlo analysis. The ADMM-RLS algorithm preserves a correct global parameters estimate only if the uncertainty around the true value is restricted, and it quickly degenerates when it grows. Algorithm 1, instead, displays a higher capability of retrieving correct estimates of the parameters across the simulations, without using any a-priori knowledge on the parameters vector.

Scenario 3

We consider a dataset of cardinality 5000 collected from system

$$\begin{aligned} y(t) = {} & 0.7y(t-1)u(t-1) - 0.5y(t-2) \\ & - 0.7y(t-2)u(t-2)^2 + 0.6u(t-1)^2 + e(t) \end{aligned}$$

with $u(t) \sim WUN(-1, 1)$, $e(t) \sim WGN(0.04)$, where $WUN(l_b, u_b)$ denotes a White Uniform Noise in the range $[l_b, u_b]$. We then partition the 5000 data among an increasing number of agents. The purpose is to assess: i) the estimation accuracy of Algorithm 1 compared with ADMM-RLS and the solution where agents learn their model independently through a L_2-regularized Least Squares based on their own data, and ii) the scalability of Algorithm 1 as the number of agents increases. We set the parameters of the ADMM-RLS algorithm as specified in Table II. The parameters are assumed to be known

TABLE I: Algorithm 1: true and average estimate of the parameters over 100 Monte Carlo runs.

Correct Selection		100%			
w_1	True value	0.7739	0.6658	0.4960	0.9413
	Average estimate	0.7740	0.6688	0.4935	0.9414
w_2	True value	0.1887	0.1192	0.2102	0.3386
	Average estimate	0.1911	0.1212	0.2096	0.3382
w_3	True value	0.6612	0.4317	0.5549	-0.2762
	Average estimate	0.6598	0.4300	0.5552	-0.2765
w_4	True value	0.2325	0.2278	0.6338	0.1345
	Average estimate	0.2339	0.2290	0.6365	0.1339

TABLE II: ADMM-RLS parameters in scenario 2.

$\hat{\theta}_n^{rls}(0)$	$\phi_n(0)$	ρ_1	ρ_2	$\delta_{n,1}^0$	$\delta_{n,2}^0$
0_5	$0.1I_5$	1	0.1	$10^{-3}I_3$	$10^{-3}I_3$

TABLE III: Parameters of Algorithm 1.

Scenario	α	γ	β	ε	δ
1	10^{-3}	10^{-3}	0.01	10^{-5}	0.9985
2	10^{-3}	1	0	10^{-5}	-
3	10^{-3}	10	0	10^{-5}	-

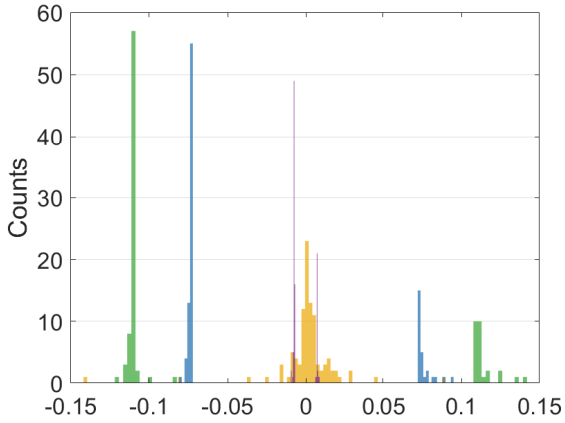

Fig. 1: Relative error distribution of the first parameter estimate for the ADMM-RLS $\pm 1\%$ (purple), ADMM-RLS $\pm 10\%$ (blue), ADMM-RLS $\pm 15\%$ (green) and Algorithm 1 (yellow).

with an uncertainty of $\pm 10\%$. The parameter setting in Table III is adopted for Algorithm 1. The weighting coefficient for the regularization term in the independent learning solution is set equal 0.001 . Table IV collects the results. Specifically, we report the average across the agents of the mean square output prediction error (MSE) over a validation dataset of the same size as the training one, and the average absolute relative error on the parameters estimates $|\overline{e}_w|$. Table IV shows how ADMM-RLS rapidly saturates to the maximum parameters uncertainty allowed by constraints. Both ADMM-RLS and the multi-agent algorithms are more robust than independent learning against the reduction of training size for the agents because of the collaborative learning scheme. Additionally, the computational robustness of our decentralized scheme with respect to the increasing number of agents involved is displayed in terms of the number of alternating algorithm iterations required on the cloud and the computational time in seconds (internal clock of the calculator).

V. CONCLUSIONS

We introduced a novel decentralized, cloud-based algorithm for nonlinear system identification in a multi-agent framework, where each agent operates on a different instance of the same system. The proposed algorithm was inspired by some developments in multi-task feature learning. Its superior performance with respect to a state-of-the-art competitor was demonstrated through some numerical examples. We are currently investigating possible application of the approach to predictive maintenance. This requires further effort.

TABLE IV: Comparative analysis in scenario 3: average values of the output prediction MSE (to be rescaled by 10^{-3}) and absolute relative error of the parameter estimate. Number of iterations and time (in seconds, to be rescaled by 10^{-2}) for Algorithm 1.

# of agents	4	8	16	32	64	128	256	512
data per agent	1250	625	312	156	78	39	19	9
	MSE							
ADMM-RLS	45	47	57	59	63	68	77	98
Algorithm 1	36	37	38	40	42	46	57	85
Ind. agents	37	37	39	40	44	52	78	237
	$\lvert\overline{e}_w\rvert$							
ADMM-RLS	7	8	10	10	10	10	10	10
Algo 1	6	6	6	5	3	5	14	40
Ind. agents	13	17	30	44	58	87	156	351
Algo 1 iter.	2	2	3	4	9	11	14	18
Algo 1 Time	6	2	2	1	1	1	2	5

REFERENCES

[1] P. Mell and T. Grance, "The NIST definition of cloud computing," Technical Report SP 800-145, National Institute of Standards & Technology Gaithersburg, MD, USA, 2011.

[2] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Delyon, P. Y. Glorennec, H. Hjalmarsson, and A. Juditsky, "Nonlinear black-box modeling in system identification: a unified overview," *Automatica*, vol. 31, no. 12, pp. 1691–1724, 1995.

[3] I. J. Lentaritis and S. A. Billings, "Input-output parametric models for non-linear systems part i: deterministic non-linear systems," *International Journal of Control*, vol. 41, no. 2, pp. 303–328, 1985.

[4] P. Palumbo and L. Piroddi, "Seismic behaviour of buttress dams: nonlinear modelling of a damaged buttress based on ARX/NARX models," *Journal of Sound and Vibration*, vol. 239, pp. 405–422, 2000.

[5] Y. Guo, L. Z. Guo, S. A. Billings, and H. L. Wei, "An iterative orthogonal forward regression algorithm," *International Journal of Systems Science*, vol. 46, pp. 776–789, 2015.

[6] A. Falsone, L. Piroddi, and M. Prandini, "A randomized algorithm for nonlinear model structure selection," *Automatica*, vol. 60, pp. 227–238, 2015.

[7] V. Breschi, A. Bemporad, and I. V. Kolmanovsky, "Cooperative constrained parameter estimation by ADMM-RLS," *Automatica*, vol. 121, p. 109175, 2020.

[8] F. Bianchi, A. Falsone, M. Prandini, and L. Piroddi, "Nonlinear system identification with model structure selection via distributed computation," in *2019 IEEE 58th Conference on Decision and Control (CDC)*, 2019, pp. 6461–6466.

[9] Y. Zhang and Q. Yang, "A survey on multi-task learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 12, pp. 5586–5609, 2022.

[10] A. Argyriou, C. Micchelli, M. Pontil, and Y. Ying, "A spectral regularization framework for multi-task structure learning." in *Advances in Neural Information Processing Systems*, vol. 20, 01 2007.

[11] A. Argyriou, T. Evgeniou, and M. Pontil, "Convex multi-task feature learning," *Machine learning*, vol. 73, pp. 243–272, 2008.

[12] L. Grippo and M. Sciandrone, "On the convergence of the block nonlinear Gauss–Seidel method under convex constraints," *Operations Research Letters*, vol. 26, no. 3, pp. 127–136, 2000.

[13] T. Evgeniou, M. Pontil, and O. Toubia, "A convex optimization approach to modeling consumer heterogeneity in conjoint estimation," *Marketing Science*, vol. 26, pp. 805–818, 11 2007.