Journal of Process Control 124 (2023) 36–53

Contents lists available at ScienceDirect

Journal of Process Control

journal homepage: www.elsevier.com/locate/jprocont

Transfer Learning in wastewater treatment plants control: Measuring the transfer suitability

Ivan Pisa [a,b,*], Antoni Morell [a], Jose Lopez Vicario [a], Ramon Vilanova [b]

[a] Wireless Information Networking (WIN) Research Group, Escola d'Enginyeria, Universitat Autònoma de Barcelona, Bellaterra, 08193, Spain
[b] Automation and Advanced Control Systems (ASAC) Research Group, Escola d'Enginyeria, Universitat Autònoma de Barcelona, Bellaterra, 08193, Spain

ARTICLE INFO

Article history:
Received 27 October 2022
Received in revised form 23 December 2022
Accepted 13 February 2023
Available online 24 February 2023

Keywords:
PID controllers
Transfer Learning
Water management

ABSTRACT

The industrial sector is nowadays experiencing a digital transformation motivated by the Industry 4.0 paradigm. Concepts such as data-driven models, Artificial Neural Networks (ANNs), and Transfer Learning (TL) are part of the current vocabulary in the industrial management and control topics. For that reason, in this paper the application of TL techniques is proposed to derive new ANN-based control structures from pre-existing ones. Notice that if an ANN-based controller is transferred into a new industrial environment, its appropriate behaviour must be ensured, and what it is more important, this must be known *a priori*. Nevertheless, TL techniques do not always ensure this. That is why the Transfer Suitability Metric (TSM) is proposed here. Determining the similarity among environments, this metric tells if the controller can be transferred, transferred with certain limitations, or if it cannot be transferred at all. Here, the metric is applied over a Wastewater Treatment Plant (WWTP). The objective is to derive the control structure of one control loop, let us say the Dissolved Oxygen (DO), and then transfer it into another basic control loop in a WWTP, the Nitrate–nitrogen (NO), and vice-versa. Results show that with the help of the TSM, an improvement around a 68.54% and 80.53% in the Integrated Absolute Error (IAE) and the Integrated Squared Error (ISE) is obtained in the NO management, respectively. Moreover, a simplification and speed-up of the controller design process is achieved.

1. Introduction

During the last years, the industrial sector has experienced a digital transformation towards the Industry 4.0 paradigm. It has arisen as one of the most important advances where industrial systems will be able to communicate among them by means of exchanging huge amounts of information. Together with the massive adoption of such systems in most of the industrial scenarios, the amount of available information related to industrial systems and their behaviour is expected to exponentially increase [1,2]. This motivates the appearance of processes and tasks where Artificial Neural Networks (ANNs) are considered as the crucial tool to carry out the desired activity. One clear example consists in the adoption of ANNs for maintenance and anomaly detection purposes, where they predict the correct behaviour of the processes under supervision and determine whether the system is behaving correctly or not [1,3]. For instance, ANNs are considered to develop soft-sensors devoted to determining the deformation produced in the rotors of air pre-heater systems [4]. Results there show that ANNs can offer a much better performance than other data-driven methods. Other recent use cases of ANNs show that Feed-forward Neural Networks (FFNN) and Bayesian ones are considered to diagnose failures. The formers, for instance, determine the damage of different structures, ranging from tiny composite plates to bigger ones such as a steel-concrete composite cantilever structure [5–7]. The latter are adopted to determine different topologies of failures of permanent magnet motors, to determine the fault diagnosis of multi-redundant closed-loop feedback control systems or to estimate the performance degradation of the system model [8–10].

Nonetheless, ANNs are not only considered to detect anomalies and malfunctions of industrial systems, but also to develop sensing tools devoted to measuring certain parameters from different environments [11]. For instance, Convolutional Neural Networks (CNNs) are considered to develop a method to track the rail head and waists by means of laser devices pointing at the desired points of the rail structure. In that way, the proposed method

* Corresponding author at: Wireless Information Networking (WIN) and Automation and Advanced Control Systems (ASAC) Research Groups, Escola d'Enginyeria, Universitat Autònoma de Barcelona, Bellaterra, 08193, Spain.
E-mail addresses: ivan.pisa@uab.cat (I. Pisa), antoni.morell@uab.cat (A. Morell), jose.vicario@uab.cat (J.L. Vicario), ramon.vilanova@uab.cat (R. Vilanova).

https://doi.org/10.1016/j.jprocont.2023.02.006

exploits the features of CNNs to provide an accurate and robust performance [12]. Other use cases of ANNs as sensing approaches are widely observed in the petrochemical and wastewater treatment sectors. For instance, a soft-sensor based on Adaptive Linear neurons is proposed to generate predictions of the densest concentration of the elements in a distillation column [13]. In [14], a soft-sensor based on recurrent neural networks, and especially Long Short–Term Memory (LSTM) cells, is proposed to determine violations of effluent values in a Wastewater Treatment Plant (WWTP).

ANNs are also considered to tackle control issues as though they were conventional controllers. For instance, in [15], two FFNNs are considered to implement the direct and inverse controller of an oil well drilling machine. Results show that this approach can regulate in real time the pressure of the drilling machine. Another example consists in the adoption of ANNs to implement an Internal Model Controller (IMC) based on Multilayer Perceptron (MLP) and FFNN networks [16]. Its main aim is to regulate the quantity of light present in an office environment. In [17], an ANN with a Proportional Integral (PI) ability is placed in the feedback loop of a control structure managing a buck DC/DC converter. In that manner, the ANN can consider a high number of gains rather than the usual number of conventional PI controllers. In [18], the power of ANNs is considered to implement a Model Predictive Controller (MPC) devoted to controlling highly complex multilevel power converters.

Focusing on the WWTP domain, ANNs are also considered as tools either supporting a control structure, or acting as the main controller. For instance, in [19], ANNs are proposed to perform predictions of future violations of the WWTP pollutant limits. These predictions are considered to activate either a fuzzy, or a MPC control structure as a function of the predicted value. In [20], ANNs are adopted in the identification process of a MPC in charge of controlling the effluent concentrations of a general purpose WWTP. Later, in [21], ANNs are considered to implement an adaptive fuzzy controller able to track the optimal set-points of the WWTP dissolved oxygen control loop. In that way, the ANN exploits its performance in the modelling of non-linear processes. As a results, a better performance compared to a conventional fuzzy controller can be achieved. Finally, in [22], Recurrent Neural Networks (RNN) are considered to implement an IMC structure in charge of the same loop. Results there show that such structures can overcome the results of conventional PI structures.

As it is observed, all these works rely on the performance of the ANN on which they are based. However, ANNs must be properly dimensioned and trained to offer a good performance. The problem here lies in the fact that the training process of an ANN can become a tedious and high time demanding and consuming task [23]. Indeed, ANNs are tools whose performance rely on the data considered in the training process. Consequently, data scarcity problems can entail severe issues. If enough data is not considered, the ANN cannot be properly trained. For all these reasons, these issues must be tackled somehow, especially in the industrial sector where ANNs are being more and more adopted.

Recent studies show that the most representative computational intelligence algorithms can be considered to alleviate these issues. For instance, a self-adaptive extreme learning algorithm is considered to determine the optimal number of neurons in an ANN structure [24]. The same algorithm is also proposed to reduce the computational requirements in the ANN training process [25]. More complex algorithms like the Monarch Butterfly Optimization process can be adopted to optimise the ANN structure and therefore, derive an optimal network offering a good performance without carrying out huge efforts [26]. The problem of the application of these algorithms in the control domain relies on the fact that they entail the derivation of a new ANN for each control loop being controlled. Consequently, their main benefit cannot be completely exploited. Here is where Transfer Learning (TL) comes into action. It is in charge of getting an ANN, which is consciously trained for a certain task in a given domain, *the source domain*, and then transferring it into a new unseen task or scenario, *the target domain*. This approach has been widely studied and adopted in classification tasks and, particularly, in image processing [27].

Notwithstanding, in the industrial sector, most of the works where TL is involved are focused on the machine fault detection and diagnosis. For instance, convolutional neural networks and TL are considered to implement a system able to manage gearbox components to avoid their failure [28]. The same principle is also observed in [29]. There, authors present a TL and fine-tuning (FT) process devoted to implementing a machine fault diagnosis system which will determine gearbox failures by means of time–frequency imaging of the gearbox system. The convolutional networks are trained with labelled data from the source domain and then transferred into the target domain where only unlabelled data is available. Finally, TL is also considered joint with the AlexNet, a neural network architecture, to obtain structures able to identify and classify different emulsion mechanisms for the oil sector [30]. Other examples considering regression tasks mainly adopt TL to implement soft-sensors to be applied in scenarios presenting data-scarcity problems [31]. In there, an ANN-based soft-sensor is trained with data from the source domain, the one where enough data are available, and then transferred into the target environment, the one showing data-scarcity issues. The same is applied in [32], where a CNN is considered to forecast the WWTP power consumption. Here, TL is proposed to overcome the issues related to the WWTP data scarcity by means of transferring pre-trained CNNs.

Moreover, in the industrial domain, the benefits of TL can be exploited to speed-up the design of control loops. For instance, TL techniques are considered in different ways to derive new control structures [33–36]. In [33], TL is applied to derive a new variable speed limit control unit. The main point is that TL is applied to introduce a reinforcement learning agent trained in the source domain into the target ones. In that manner, an acceleration in the training process and the control design is achieved. In [34], TL is considered to transfer a deep reinforcement learning agent managing the heating system of a source building into a target one. Thus, the new heating system can be obtained in an easy manner. Later, in [35], TL accelerates the training process of a hybrid cloud and edge reinforcement learning based control strategy. Finally, in [36], transfer learning is considered to derive new reinforcement learning based controllers of heating pumps by means of transferring the controllers of similar systems in the same environment.

In terms of the WWTP management, a first assessment of the application of TL to derive new controllers was presented in [37,38], where an ANN-based control structure is transferred among the different control loops. Results there show that a reduction of the ANN design and training time is achieved. However, the adoption of TL in such a way entails some issues that must be considered:

- TL is not always a workable solution, and, on some occasions, the Negative Transfer effect can be suffered [39].
- In industrial scenarios, TL cannot be evaluated *a posteriori* since a wrong actuation can entail the destruction of the target industrial environment.
- There exists a lack of metrics able to show the suitability of transferring the ANN-based control design approach.

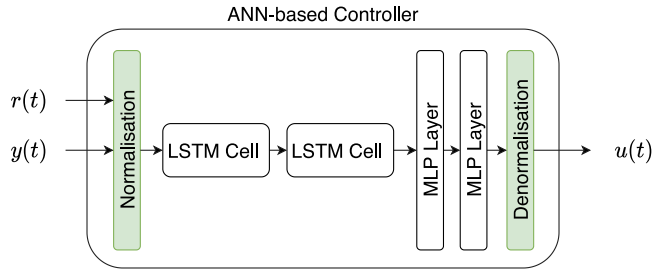


Fig. 1. ANN-based Controller considering two LSTM cells and two MLP layers. Its main aim is to replicate the behaviour of a conventional control structure.

For that reason, this paper tries to shed some light in this domain providing the industrial control developers with a new metric measuring the transfer suitability of ANN-based controllers. Notice that this paper corresponds to an extension of the works shown in [37,38], where all the aforementioned issues were not considered. In that sense, this work firstly proposes and defines this metric conceptually and mathematically. Later, its behaviour is analysed considering four different academic examples. Finally, it is considered and applied in the development of new ANN-based control structures adopted in the same domain as in [37,38], a WWTP environment.

The outline of the paper is as follows: Section 2 defines the problem being tackled in this paper. The proposed metric solving the problem previously defined is presented in Section 3. Its validation is attained in Section 4 and its final application is performed in Section 5. Finally, Section 6 concludes the paper.

2. Problem definition

Industrial Environments are characterised by presenting huge amounts of control loops devoted to ensuring the correct behaviour and functionality of the industrial plant. In that sense, the amount of time required to implement and design each one of these controllers increases with the number of control loops to manage. Notwithstanding, TL arises as an option able to reduce this design and implementation time [38]. To exploit the benefits of TL, an ANN-based Controller (see Fig. 1) must be designed and properly trained so as to correctly manage an industrial process, *the source environment*. Later, this ANN-based controller will be deployed on the remaining loops or *target environments*, not only reducing the design complexity, but also the required time to derive a control solution [38]. Here, it consists in two LSTM cells and two MLP layers connected in series. The number of hidden units per layer equals to 100, 50, 25 and 1, respectively. LSTM cells are considered here due to their power in the management of time-series signals such as the ones managed in control problems [40].

Nevertheless, the ANN-based controller cannot be transferred without determining *a priori* its suitability and preventing a Negative Transfer [39]. Its effects can entail the malfunction of the control loop and therefore, the incorrect operation of the industrial process. The avoidance of this effect is required for any industrial scenario, but it is indispensable for critical industries such as WWTPs and petrochemistry environments, where a malfunction or mismanagement can entail the destruction of the industrial plant or the environment. For that reason, the need of measuring the suitability of TL is obvious. Some works have addressed this in the last years. In [41], a first attempt to determine how transferable are certain parts of neural networks has been performed. In there, the effects of TL and fine-tuning processes performed to a set of neural networks dealing with the ImageNet dataset are assessed. Something similar is done in [42], where the limits of transfer learning are studied from the theoretical point of view. From a practical one, in [43], TL suitability is determined training several convolutional neural networks in the source domain. Then, they are transferred into the target domain and ranked with data coming from the source environment. The best network is finally considered as the model to be used. In [44], the adoption of several metrics is evaluated to determine the TL effect *a posteriori*. Nine different non-intrusive load monitoring algorithms are tested with source and target data to classify and estimate the energy consumption. However, results are only proved on energy consumption datasets, which cannot assure their adoption in other environments. From the regression point of view, Boudabous et al. present a traffic flow estimation system where its transferability is assessed [45]. Another example is shown in [46], where the transferability and accuracy of ANNs in charge of estimating the soil moisture is analysed. There, the transferability analysis is performed to determine the best dataset to train an ANN able to perform the desired prediction task. Finally, in [47], a metric to quantify the separability of data classes in classification issues is presented.

Although these works present an attempt to measure the transfer suitability of the different structures, all of them show the same issues:

- Most of the measuring approaches are devoted to determining the transfer suitability of structures performing classification tasks. However, here we are dealing with a regression problem since we want to transfer an ANN-based controller dealing with time-series signals. Therefore, the approaches for classification problems are not valid since the time correlation, which is a common feature of industrial measurements, is broken.
- The transferability of the structures dealing with regression tasks is usually computed *a posteriori* [37,38,46]. Thus, bad actuations and performances cannot be avoided.

Taking all this into account, this work is focused on providing a new measuring approach, the Transfer Suitability Metric (TSM), which can *a priori* determine the transfer suitability of an ANN-based control structure. Its main aim is to determine the similarity of the control loops between environments. Thus, the TSM metric, which is based on the idea presented in [47] and the correlations presented in [46], is computed to give a numerical value to this similarity and therefore, to decide if the ANN-based structure is transferable or not. Fig. 2 depicts the idea behind the problem that TSM is willing to solve.

A first approach to this behaviour is proposed in [48], where the first definition of the metric was proposed. However, it was tested *a posteriori* over academic examples. Here, we provide a deeper understanding of the proposed metric. Besides, we propose the adoption of the metric over academic examples and a digital version of a critical industrial environment, a WWTP. In both cases the metric is computed *a priori* considering two different control configurations: (i) an open loop one where the environment is not being controlled and (ii) considering the historical data of the closed loop. As a summary, the main contributions of this work are:

- Reduce the complexity and speed-up the control design process by means of transferring ANN-based structures between industrial environments.
- Define and develop a metric able to compute the suitability of transferring an ANN-based controller between critic source and target control loops. The key point here is that this metric is computed *a priori*. This is crucial since one can determine the performance of the controller in the target domain before performing the transference.

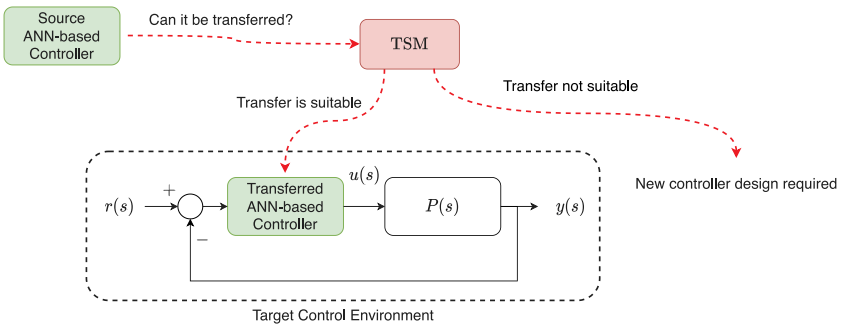


Fig. 2. TSM objective. The main idea is to determine if the ANN-based controller derived from the source domain can be transferred into the target domain or not. Notice that $r(s)$, $P(s)$, $u(s)$ and $y(s)$ are the reference signal, the process under control, and the actuation and controlled signals, respectively.

- Assess the behaviour of the metric in a critical environment such as WWTP domain.

The fact that this metric is computed *a priori* opens a new paradigm that can be highly beneficial in the industrial control design: industrial controllers can be implemented by means of ANNs which are able to replicate the behaviour of conventional structures such as Proportional Integral Derivative (PID) controllers, MPCs or IMCs [19,20,22,49,50]. They will be designed for the source scenario and then, tested with measurements from the target one. Measurements of the source domain are already obtained since they are considered in the training process of the ANN-based controller. On the other hand, the ones coming from the target environment can be obtained directly from historical data. Hence, the transference suitability of the control structure can be assessed without disturbing the target environment performance.

3. Transfer suitability metric

3.1. TSM introduction

TSM is focused on computing the similarity between the source and target environments where the transference of the ANN-based control structure will be performed. Since industrial measurements from both domains can be understood as time-series signals, the computation of correlations is proposed to determine these similarities [46]. The main idea is to obtain the outputs of the ANN-based controller when it is dealing with measurements from the source and target domains, respectively. Then, the correlations are computed to determine if the outputs are similar enough to consider the transference of the ANN-based controller. High correlation values will be obtained in those situations where the ANN is treating the measurements from both domains similarly and therefore, when the ANN-based controller is transferable. On the other hand, low values will be obtained in those situations where the ANN is performing different processes to the input measurements. Consequently, the control problems are different enough to not apply the ANN-based controller transference. In that sense, the autocorrelation and cross-correlation between measurements of both domains are adopted here.

3.2. TSM definition

In order to derive the TSM metric, let us consider two different processes, a First Order Plus Dead-Time (FOPDT) process, $P_s(s)$, and a Second Order Plus Dead-Time (SOPDT) process, $P_t(s)$. Both have been previously defined and deeply assessed in [51,52]:

$$P_s(s) = \frac{K}{Ts+1}\cdot e^{-Ls} = \frac{1.4}{1.2s+1}\cdot e^{-0.4s}$$
$$P_t(s) = \frac{K\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}\cdot e^{-Ls} = \frac{1.521}{s^2+0.338s+1.59}\cdot e^{-0.370s} \tag{1}$$

where K, T and L correspond to the gain, time constant, and delay of the FOPDT process, respectively. In terms of the SOPDT, the ω_n equals to the natural frequency of the process and ζ to its damping coefficient.

In such a context, $P_s(s)$ is controlled by an ANN-based PID controller such as the one presented in Fig. 1. It is properly trained so as to replicate the PID controller defined in [51]. Then, it is proposed to be transferred into $P_t(s)$. Hence, the similarity between industrial processes is computed to determine if $P_t(s)$ can be controlled by the same ANN-based controller.

Consequently, two matrices are obtained for each one of the ANN-based layers when dealing with measurements from $P_s(s)$ and $P_t(s)$, respectively. These matrices are defined as $\mathbf{X}^i, \mathbf{Y}^i \in \mathbb{R}^{N\times D}$, where D is the output dimension of the ith layer of the ANN structure and N is the number of time instants (see Fig. 3). Thus, $\mathbf{X}^i$ and $\mathbf{Y}^i$ can be understood as the set of outputs of the ith layer sorted in time, i.e., $\mathbf{X}^i = \left[\mathbf{x}^i(t_0), \ldots, \mathbf{x}^i(t_N)\right]^T$ and $\mathbf{Y}^i = \left[\mathbf{y}^i(t_0), \ldots, \mathbf{y}^i(t_N)\right]^T$, where $\mathbf{x}^i(t), \mathbf{y}^i(t) \in \mathbb{R}^{1\times D}$ are the outputs of the ith layer at a time instant $t \in [t_0, t_N]$.

The first step to compute the proposed metric is to reduce the dimension of each ANN layer to a determined one. This is performed to obtain an invariant metric with respect to the dimensions of each layer of the ANN-based controller and to preserve their time correlation. The number of dimensions can be set to whatever dimension is desired, however, in this work a 2-dimension matrix is proposed since it allows for a visual interpretation. Here, the algorithm to perform this task consists in the well-known Multidimensional Scaling (MDS) algorithm [53,54]. While other possible methods such as Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and Independent Component Analysis (ICA) reduce the dimension of data without preserving the relationships among measurements, the MDS algorithm is considered due to its ability to preserve these relationships once the dimensions are reduced [53]. It is important to preserve them since later the correlation computation will take them into account. In that sense, the following transformation is performed:

$$\begin{aligned}\mathbf{X}^i \in \mathbb{R}^{N\times D} &\xrightarrow{MDS} \mathbf{X}'^{,i} \in \mathbb{R}^{N\times 2}\\ \mathbf{Y}^i \in \mathbb{R}^{N\times D} &\xrightarrow{MDS} \mathbf{Y}'^{,i} \in \mathbb{R}^{N\times 2}\end{aligned} \tag{2}$$

Visually, one can observe in Fig. 4 that the output of the layer is changed from a set of values in the form of a matrix, which

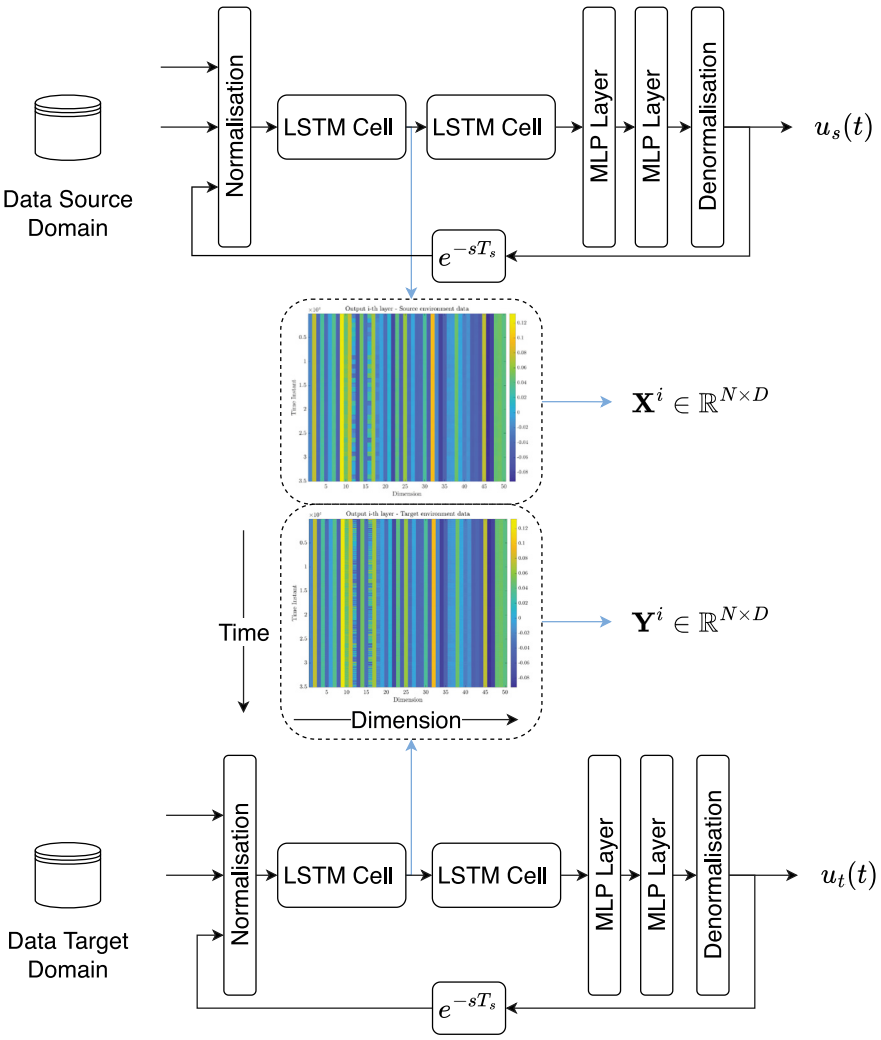


Fig. 3. Structure of matrices $\mathbf{X}^i$ and $\mathbf{Y}^i$ for the ith layer of the ANN-based controller. Notice that a continuous time-delay block is added with a delay of T_s. In this case, T_s corresponds to the sampling time of measurements which holds the Nyquist Sampling Theorem.

cannot be easily managed, to a situation where the output of the layer can be easily interpreted as the evolution of a signal in a 2-dimensional space. For instance, we are going to take the outputs of the second LSTM layer (LSTM 2) of the ANN-based controller presented in [38], not only when it is controlling $P_s(s)$, but also $P_t(s)$. In this case, it can be clearly observed that the evolution of the dimensions of the ANN-based controller are not similar at all.

Since the correlation values highly depend on the input measurements, the time-correlation matrices consist in the normalised autocorrelation of the data coming from the source control loop, $\mathbf{R}_{x',i} \in \mathbb{R}^{(2N-1)\times 2}$, and the normalised cross-correlation between the data coming from the source and the target control loops, $\mathbf{R}_{x'y',i} \in \mathbb{R}^{(2N-1)\times 2}$. Thus, they consist in:

$$\mathbf{R}_{x',i} = \begin{bmatrix} \frac{1}{\max(\mathbf{r}_{x'^i_{j=1}})} \cdot \mathbf{r}_{x'^i_{j=1}} \\ \frac{1}{\max(\mathbf{r}_{x'^i_{j=2}})} \cdot \mathbf{r}_{x'^i_{j=2}} \end{bmatrix} \tag{3}$$

$$\mathbf{R}_{x'y',i} = \begin{bmatrix} \frac{1}{\sqrt{\max(\mathbf{r}_{x'^i_{j=1}})\max(\mathbf{r}_{x'y'^i_{j=1}})}} \cdot \mathbf{r}_{x'y'^i_{j=1}} \\ \frac{1}{\sqrt{\max(\mathbf{r}_{x'^i_{j=2}})\max(\mathbf{r}_{x'y'^i_{j=2}})}} \cdot \mathbf{r}_{x'y'^i_{j=2}} \end{bmatrix} \tag{4}$$

where $\mathbf{r}_{x'^i_j}$ and $\mathbf{r}_{x'y'^i_j} \in \mathbb{R}^{(2N-1)\times 1}$ are the sample autocorrelation and cross-correlation vectors for $\left[\mathbf{X}'^{,i}\right]_{j,\cdot}$ and $\left[\mathbf{Y}'^{,i}\right]_{j,\cdot}$, respectively.

Once the normalised autocorrelation and cross-correlation matrices for the given layer of the ANN-based controller are obtained, we proceed with the computation of the TSM for the ith layer:

$$\mathrm{TSM_i} = 1 - \sqrt{\frac{1}{2(2N-1)} \sum_{k=1}^{2N-1} \sum_{j=1}^{2} \left(\left[\mathbf{R}_{x',i}\right]_{k,j} - \left[\mathbf{R}_{x'y',i}\right]_{k,j} \right)^2} \tag{5}$$

Then the TSM value for the considered ANN-based controller is computed as:

$$\mathrm{TSM} = min(\mathrm{TSM_i}) \tag{6}$$

Taking all this into account, the TSM computation can be performed following the Algorithm 1.

Notice that TSM relies on data, which may come from SCADA systems as well as physical sensors that might introduce noise to the measurements or return missing values. In such a context, prior to compute the TSM a data preprocessing stage should be performed to assure that measurements corrupted by noise are cleaned and missing values are imputed. Data-based strategies such as the application of Autoencoders and Denoising Autoencoders can be applied [22,55]. In addition, we have empirically proved that only a total amount equivalent to the 30% of the required data to properly train the models is needed to safely compute the TSM value in the scenarios proposed in this work.

Algorithm 1 Algorithm to compute the TSM metric

for $i = 1$ **to** M **do** ▷ $M \leftarrow$ Number of layers

Obtain the data:

$\mathbf{X}^i$ = outputs of the i-th ANN layer controlling $P_s(s)$ ▷ $\mathbf{X}^i \in \mathbb{R}^{N\times D}$

$\mathbf{Y}^i$ = outputs of the i-th ANN layer controlling $P_t(s)$ ▷ $\mathbf{Y}^i \in \mathbb{R}^{N\times D}$

Compute the MDS:

$\mathbf{X}'^{,i} = MDS(\mathbf{X}^i, 2)$ ▷ $\mathbf{X}'^{,i} \in \mathbb{R}^{N\times 2}$

$\mathbf{Y}'^{,i} = MDS(\mathbf{Y}^i, 2)$ ▷ $\mathbf{Y}'^{,i} \in \mathbb{R}^{N\times 2}$

for $j = 1$ **to** 2 **do**

Define $\mathbf{X}_j^{\Delta,i}$:

$$\mathbf{X}_j^{\Delta,i} = \begin{bmatrix} [\mathbf{X}'^{,i}]_{N,j} & 0 & \cdots & 0 \\ [\mathbf{X}'^{,i}]_{N-1,j} & [\mathbf{X}'^{,i}]_{N,j} & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ [\mathbf{X}'^{,i}]_{1,j} & [\mathbf{X}'^{,i}]_{2,j} & \cdots & [\mathbf{X}'^{,i}]_{N,j} \\ 0 & [\mathbf{X}'^{,i}]_{1,j} & \cdots & [\mathbf{X}'^{,i}]_{N-1,j} \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & [\mathbf{X}'^{,i}]_{2,j} \\ 0 & 0 & \cdots & [\mathbf{X}'^{,i}]_{1,j} \end{bmatrix}$$

▷ $\mathbf{X}_j^{\Delta,i} \in \mathbb{R}^{(2N-1)\times N}$

Compute autocorrelation $\mathbf{r}_{x'_j^i}$ and cross-correlation $\mathbf{r}_{x'y'_j^i}$ vectors:

$\mathbf{r}_{x'_j^i} = \mathbf{X}_j^{\Delta,i} \cdot [\mathbf{X}'^{,i}]_{\cdot,j}$ ▷ $\mathbf{r}_{x'_j^i} \in \mathbb{R}^{(2N-1)\times 1}$

$\mathbf{r}_{x'y'_j^i} = \mathbf{X}_j^{\Delta,i} \cdot [\mathbf{Y}'^{,i}]_{\cdot,j}$ ▷ $\mathbf{r}_{x'y'_j^i} \in \mathbb{R}^{(2N-1)\times 1}$

Normalise autocorrelation and cross-correlation:

$[\mathbf{R}_{x'^{,i}}]_{\cdot,j} = \frac{1}{\max(\mathbf{r}_{x'_j^i})} \cdot \mathbf{r}_{x'_j^i}$

$[\mathbf{R}_{x'y'^{,i}}]_{\cdot,j} = \frac{1}{\sqrt{\max(\mathbf{r}_{x'_j^i})\cdot\max(\mathbf{r}_{x'y'_j^i})}} \cdot \mathbf{r}_{x'y'_j^i}$

end for

Compute Transfer Suitability Metric for i-th ANN layer:

$$\mathrm{TSM}_i = 1 - \sqrt{\frac{1}{2(2N-1)} \sum_{k=1}^{2N-1} \sum_{j=1}^{2} \left([\mathbf{R}_{x'^{,i}}]_{k,j} - [\mathbf{R}_{x'y'^{,i}}]_{k,j} \right)^2}$$

end for

Compute Transfer Suitability Metric as:

$\mathrm{TSM} = min(\mathrm{TSM}_i)$

If lower amounts are considered, however, some dynamics of the ANN-based controller management can be lost since the slowest dynamics could not be properly captured. Besides, the TSM could not converge efficiently due to the lack of this information and therefore, produce undesired outputs.

3.3. TSM interpretation

Once the minimum TSM value is computed, the rule to decide between transferring or not is the following one:

$$\mathrm{Transfer} = \begin{cases} \text{Suitable} & \text{if TSM} > \gamma_{suitability} \\ \text{Unsuitable} & \text{if TSM} \leq \gamma_{suitability} \end{cases} \quad (7)$$

where $\gamma_{suitability}$ corresponds to the transfer suitability limit.

It is worth noting that, with the considered normalisation, the maximum and minimum correlation values are set to 1 and 0, respectively. For that reason, values close to 1 mean that the autocorrelation and cross-correlation are remarkably similar and therefore, that the ANN-based controller can behave similarly in the source and the target domain. Otherwise, TSM values equal to 0 would be obtained in the hypothetical scenario where the autocorrelation matrix totally differs from the cross-correlation matrix.

To clearly observe that, let us consider the case of transferring the ANN-based controller of $P_s(s)$ into $P_t(s)$. TSM values are computed for each layer, showing that its maximum is given at the output of the ANN-based controller. It equals to 0.665. On the other hand, the lowest TSM, which equals to 0.5683, is offered by the first MLP layer. Visually, Fig. 5 clearly shows that the transference of the ANN-based controlling $P_s(s)$ into $P_t(s)$ is not a suitable option. From this, it can be extrapolated that TSM values closer to 0.6 or lower will be obtained in those cases where the transfer of the ANN-based controller is not suitable. For that reason, and to leave a certain margin, the transfer suitability limit is set to $\gamma_{suitability} = 0.7$. Notice that this value is ad-hoc. Therefore, $\gamma_{suitability}$ should be computed for each domain, i.e., other control scenarios. However, a rule-of-thumb has been considered to set the value of $\gamma_{suitability}$: it must be placed above 0.7. TSM is based on the correlation concept and values higher than 0.7 are considered as strong correlations [56]. From that minimum value, the higher the value of $\gamma_{suitability}$, the stronger the required relationship between domains (which could be the case when addressing more complex and critical processes).

4. TSM validation

Before applying the TSM on the WWTP environment, its behaviour is assessed considering four academic controlled processes (see Fig. 6). They consist in two FOPDT systems, $P_s(s)$ and $P_2(s)$, an over-damped SOPDT process, $P_3(s)$, and a SOPDT system with resonant frequencies, $P_4(s)$. $P_s(s)$ is considered as the source domain while $P_2(s)$, $P_3(s)$, and $P_4(s)$ are considered as the target ones. All these systems have been previously developed and deeply analysed by Kurokawa et al. [51,52].

4.1. Source environment

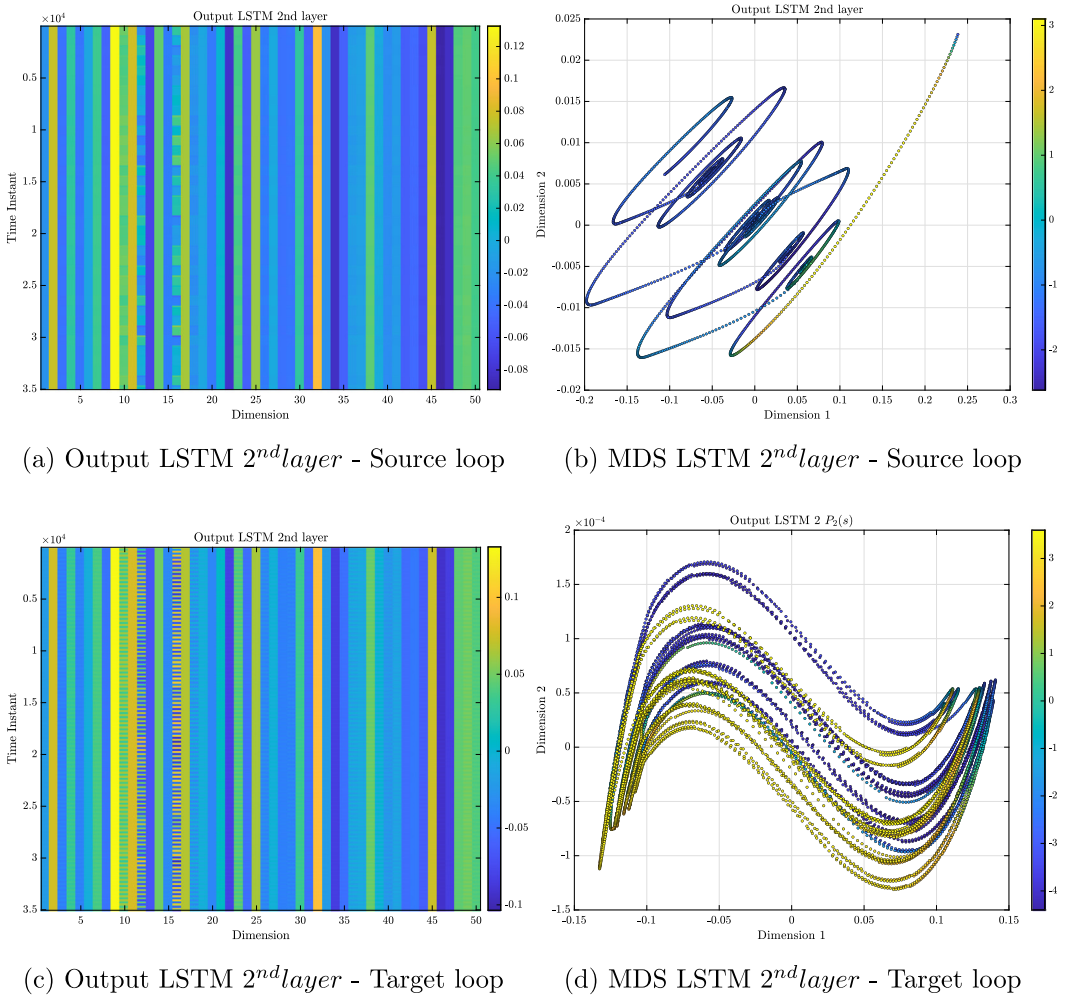
$P_s(s)$ is defined in (1), where its gain, K, is equal to 1.4 and its dead-time or delay to $L = 0.4$ s. The conventional one degree of freedom PID controller controlling this process is taken from Kurokawa et al. in [51]. It is defined as:

$$u_s(s) = 1.286 \cdot \left[\left(1 + \frac{1}{1.8109s} \right) \cdot e_s(s) + \frac{0.2367s}{\alpha \cdot 0.2367s + 1} \cdot y_s(s) \right] \quad (8)$$

where $e_s(s)$ corresponds to the mismatch between the reference signal $r_s(s)$ and the output signal of $P_s(s)$, $y_s(s)$.

Taking all this into account, the signals of the $P_s(s)$ control system ($y_s(s)$ and $u_s(s)$) when it is controlled by the Kurokawa's PID are considered in the training process of the ANN-based PID controller. It is trained considering a variable set-point, $r(s)$, which is generated by means of a pseudo-random uniformly distributed sequence of values between 0 and 2.

The ANN-based PID structure is developed and trained in TensorFlow in its 1.14 version over Python 3.6 [57]. The most common libraries of Python, such as Pandas, Numpy, Matplotlib, Scikit-Learn and Scipy are also considered [58–62]. Finally, the ANN is trained considering the ADAM optimiser and back-propagation training algorithm with a learning rate of $1 \cdot 10^{-3}$ and a L_2 penalty of $1 \cdot 10^{-4}$ to avoid overfitting issues [40,



(a) Output LSTM $2^{nd}layer$ - Source loop (b) MDS LSTM $2^{nd}layer$ - Source loop

(c) Output LSTM $2^{nd}layer$ - Target loop (d) MDS LSTM $2^{nd}layer$ - Target loop

Fig. 4. D-dimensional vs. 2-dimensional outputs of the ANN-based controller when it is managing the source and target control loops.

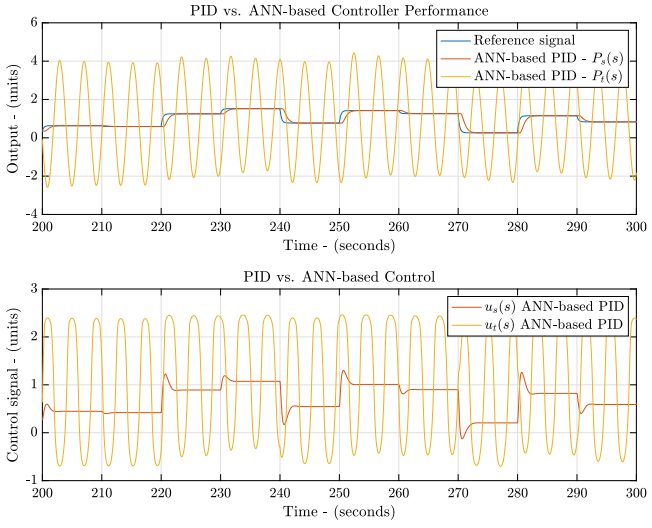


Fig. 5. Control performance when $P_s(s)$ and $P_t(s)$ systems are managed with the ANN-based controller.

Section 7.2]. This process is carried out considering an NVIDIA GeForce RTX 2080 Ti GPU and an Intel Xeon CPU working at a maximum clock frequency of 3.60 GHz. Once trained, the ANN-based PID controller is implemented over Matlab® R2020b and Simulink® v.10.2. For further details readers are referred to [38].

Results show a Root Mean Squared Error (RMSE) equal to 0.0630 units, a Mean Average Percentage Error (MAPE) of 8.57%, and a determination coefficient (R^2) of 0.996. Details on the prediction performance metrics can be observed in [63, Performance evaluation].

Then, the ANN-based structure is tested as the $P_s(s)$ controller to assess its control behaviour as well as to determine if it can be considered as a candidate to be transferred. This performance is evaluated by means of the usual Integrated Absolute Error (IAE) and Integrated Squared Error (ISE) indexes. Results show that the IAE and ISE values for $P_s(s)$, when it is controlled by the conventional PID of (8) for a variable set-point during 365 s, are equal to 16.6027 and 6.8499, respectively. In terms of the ANN-based controller, these amounts are increased until 18.6457 and 7.0470, respectively, which are only deviated a 12.03% and a 2.88% with respect to the conventional PID results, respectively. Besides, it can be observed in Fig. 7 that the control performance and the manipulated signal of the proposed ANN-based controller almost match the conventional PID as it is expected.

4.2. Target environments

In terms of the target environments, the second FOPDT process, $P_2(s)$, is defined as follows:

$$P_2(s) = \frac{2.5}{1.4s + 1} e^{-0.6s} \tag{9}$$

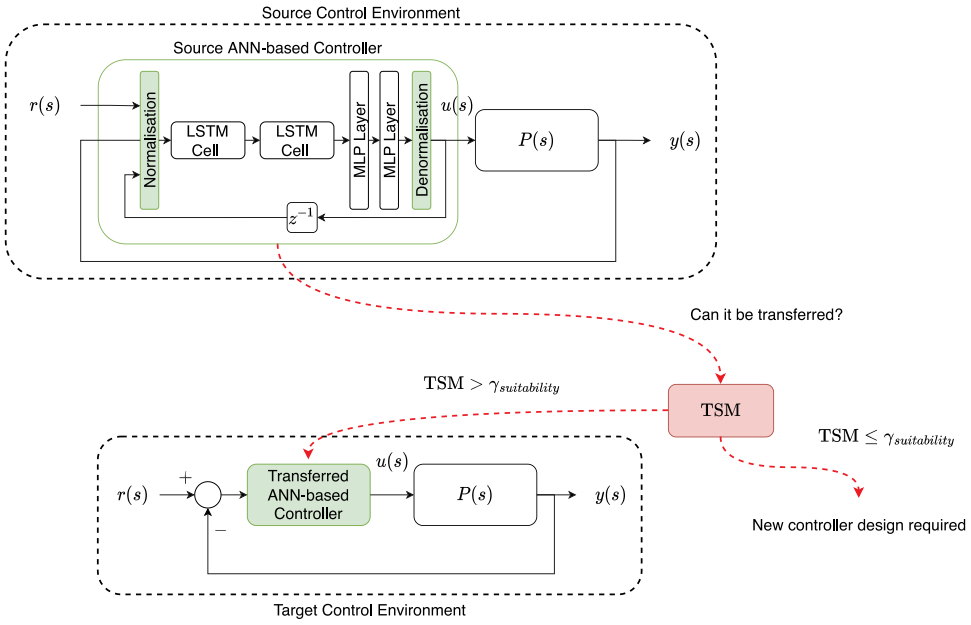


Fig. 6. TSM validation. The main idea is to determine if the ANN-based controller for the source domain can be transferred over the target domains.

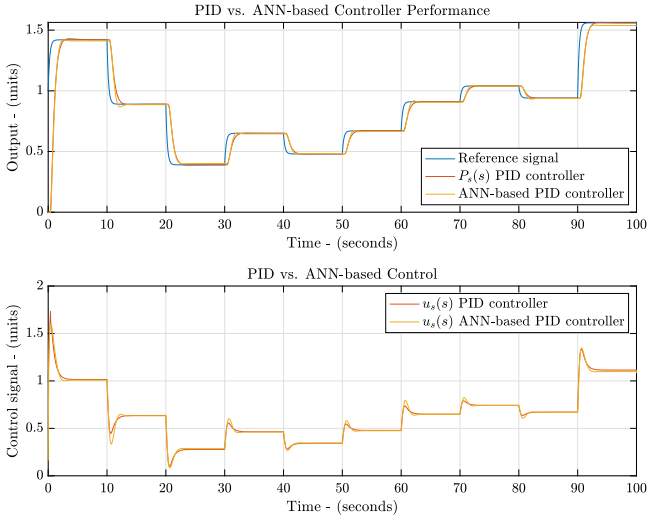


Fig. 7. Control performance when $P_s(s)$ is managed with the conventional PID and the ANN-based controller.

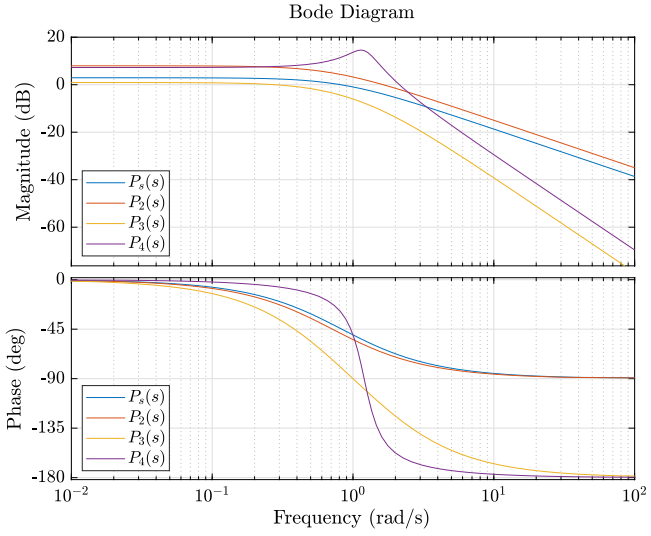


Fig. 8. Bode diagram for $P_s(s)$, $P_2(s)$, $P_3(s)$ and $P_4(s)$ processes.

where $K = 2.5$, $T = 1.4$ and $L = 0.6$ s. As it is observed, the dynamics of $P_2(s)$ differ from $P_s(s)$ mainly in the introduced delay as well as in the gain of the process.

The first SOPDT process, $P_3(s)$, is described by the following transfer function,

$$P_3(s) = \frac{1.11}{s^2 + 2.2200s + 1} \cdot e^{-0.420s} = \frac{1}{s + 1.5918} \cdot \frac{1.11}{s + 0.6282} \cdot e^{-0.420s} \tag{10}$$

which defines an over-damped process with $\zeta = 1.11$. On the other hand, $P_4(s)$ shows a resonance phenomenon at 1.14 rad/s. This phenomenon can be observed in the Bode Diagram where the gain of the process increases instead of decreasing or being constant along the frequencies (see Fig. 8). Thus, $P_4(s)$ equals to:

$$P_4(s) = \frac{3.3120}{s^2 + 0.5280s + 1.4400} \cdot e^{-0.310s} \tag{11}$$

where $\omega_n = 1.2$, $\zeta = 0.22$, and $K = 2.3$ [52].

As it has been previously stated, the main objective of this work is to compute the suitability of transferring an ANN-based controller between different domains. For that reason, the ANN-based PID controlling $P_s(s)$ is transferred into $P_2(s)$, $P_3(s)$ and $P_4(s)$ control loops. In that sense, one can face two different configurations of the target processes:

1. The processes present an open-loop configuration and therefore, data from the target domain does not contemplate any kind of information regarding the control behaviour.
2. Historical data from the target domain being managed by default PID controllers are available.

4.2.1. Open-loop data-generation scenario

The first configuration corresponds to the open-loop one, where target processes are not being controlled. Therefore, information regarding the control effects over the processes are not

Table 1
TSM between correlations when transferring the $P_s(s)$ data-based controller to manage $P_{i,2}(s)$, $P_{i,3}(s)$ and $P_{i,4}(s)$ systems.

TSM from $P_s(s)$ to $P_{i,2}(s)$, $P_{i,3}(s)$ and $P_{i,4}(s)$ systems						
Layers	$P_s(s)$ to $P_{i,2}(s)$		$P_s(s)$ to $P_{i,3}(s)$		$P_s(s)$ to $P_{i,4}(s)$	
	Fix	Variable	Fix	Variable	Fix	Variable
Input	0.7823	0.9581	0.8803	0.9739	0.6954	0.7873
LSTM 1	0.8014	0.9475	0.9003	0.9718	0.7357	0.6889
LSTM 2	0.8300	0.9447	0.9334	0.9717	0.6084	0.6857
MLP 1	0.8486	0.9478	0.9345	0.9721	0.6289	0.6450
MLP 2	0.7143	0.9789	0.8850	0.9867	0.4713	0.5909
Minimum	0.7143	0.9447	0.8803	0.9717	0.4713	0.5909
Computation Time - (s)	33.28	161.77	37.02	166.01	33.85	162.82

available. One option tackling this could be to directly place the ANN-based PID over the desired target environment, but this is counterproductive, especially for critical infrastructures since no information about the control performance is known.

To alleviate this, the identification of $P_2(s)$, $P_3(s)$ and $P_4(s)$ is performed to estimate the ANN-based PID behaviour. Here, we opt for the 123c identification model [64]. Notice that SOPDT processes are identified by Double Pole Plus Dead-Time (DPPDT) models:

$$P(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \cdot e^{-Ls} \rightarrow P'(s) = \frac{K'}{(T's+1)^2} e^{-L's} \tag{12}$$

Once identified, the models can be adopted to close the control loop and therefore, determine the behaviour of the ANN-based PID without testing it in the final industrial environment. In such a context, the identification process performed over the target scenarios provides the following process models:

$$\begin{aligned} P_{i,2}(s) &= \frac{2.5}{1.399s+1} \cdot e^{-0.6s} \\ P_{i,3}(s) &= \frac{1.11}{(1.1410s+1)^2} \cdot e^{-0.3240s} \\ P_{i,4}(s) &= \frac{2.300}{(0.3468s+1)^2} \cdot e^{-1.3040s} \end{aligned} \tag{13}$$

where $P_{i,2}(s)$, $P_{i,3}(s)$ and $P_{i,4}(s)$ stand for the identifications of $P_2(s)$, $P_3(s)$ and $P_4(s)$, respectively. $P_2(s)$ and $P_{i,2}(s)$ are nearly equal since $P_2(s)$ is a FOPDT system which can be easily identified.

The results of the identification process can be observed in Fig. 9, where the identified processes are contrasted to the original ones when a unitary step is considered as the input. As it is observed, the identified systems yield values really close to the ones from the original processes except for $P_{i,4}(s)$. This fact is motivated by the identification process as such. Since SOPDT processes are modelled by a DPPDT, the effects of resonant frequencies are lost. Therefore, the transient state of the identified model differs from the original one to later be as close as possible in the steady state.

Results concerning the transferability of the ANN-based PID into the target environments with an open-loop configuration are observed in Table 1. TSM is computed for the outputs of each ANN-based PID's layer when it is controlling the identified versions of the target environments, i.e., $P_{i,2}(s)$, $P_{i,3}(s)$, and $P_{i,4}(s)$. A fix set-point equal to one unit and a variable set-point are considered in the evaluation of the ANN-based PID control performance. Hence, it is clearly observed that the transference from $P_s(s)$ to $P_{i,4}(s)$ is not feasible at all since none of the considered set-points provide a minimum TSM over the transference limit. For instance, the minimum TSM for a variable set-point equals to 0.5909 whereas for a fixed one it equals to 0.4713.

On the other hand, the transference from $P_s(s)$ to $P_{i,3}(s)$ is recommendable. In this case, all the layers of the ANN-based PID return TSM values placed over the suitability limit. In terms of the fix set-point, the lowest TSM is given by the input layer, where the ANN-based PID does not actuate over the measurements. TSM values for the variable set-point increase until showing a minimum TSM equal to 0.9717. Now, the highest value is obtained at the output of the MLP2 layer, showing that the actuation signals for $P_s(s)$ and $P_{i,3}(s)$ are really similar. Finally, TSM values regarding the transference from $P_s(s)$ to $P_{i,2}(s)$ show that this transference can be performed as well. The minimum TSM for a variable set-point equals to 0.9447, clearly showing that the proposed ANN-based PID can be transferred. However, the minimum TSM value for a fix set-point diminishes until a TSM equivalent to 0.7143. This shows that the control behaviour of the ANN-based PID managing $P_{i,2}(s)$ and a fix set-point can be compromised (see Fig. 10).

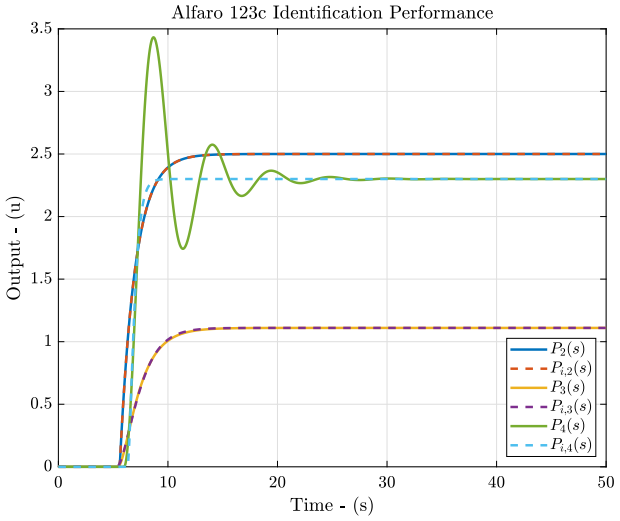


Fig. 9. Target processes identified models.

Regarding the required time to compute the TSM, it is provided as the Computation Time. In this case, for a total amount of 10 000 samples, it is observed that the highest Computation Time is obtained when a variable set-point is considered. In average, this time equals to 163.53 s. For a fix set-point, it is equivalent to 34.72 s in average, which represents a decrement of 128.82 s. This fact is motivated by the dynamics of the set-point. As its name indicates, the fix set-point consists in a constant value generating a constant output, whereas the variable set-point considers different input and output dynamics. Consequently, these dynamics are more difficult to process by the MDS algorithm and therefore, require an extra computational time.

4.2.2. Closed-loop data-generation scenario

The other configuration that one may face corresponds to the closed-loop one, where conventional control structures are

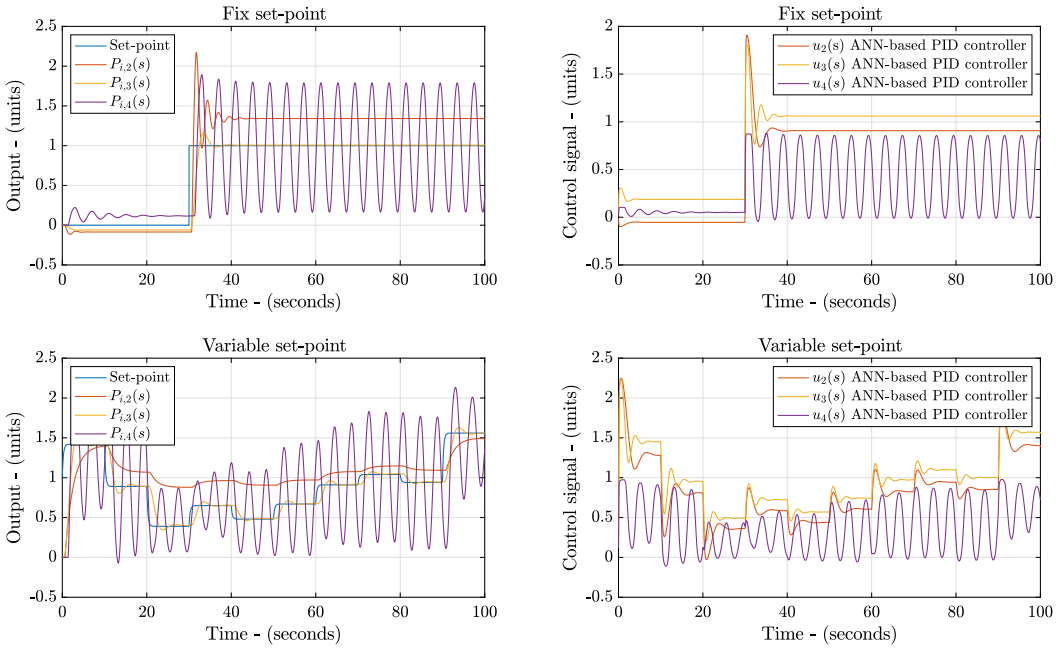


Fig. 10. Control performance when $P_s(s)$ is transferred into $P_{i,2}(s)$, $P_{i,3}(s)$ and $P_{i,4}(s)$.

Table 2
Parameters of the conventional PIDs for $P_2(s)$, $P_3(s)$ and $P_4(s)$.

Controller	K_{PID}	$T_{i,PID}$	$T_{d,PID}$
$u_2(s)$	0.6371	1.8109	0.2367
$u_3(s)$	2.6200	3.3800	0.4400
$u_4(s)$	0.2780	0.6760	1.6200

already controlling the target environments. In that sense, TSM cannot be computed by means of substituting the conventional structures by the ANN-based PID. This could entail drastic issues since the correct behaviour of the ANN-based structure is not ensured for the target domain. Instead, the adoption of historical data is considered. These data will be directly obtained from historic measurements without the necessity of generating neither new control structures, nor identifying the process under control. Nevertheless, some deviations with respect to the real behaviour of the ANN-based PID over the target scenario can appear.

Historical data consist in the input and output measurements obtained from the actuation of the conventional PID structures defined in [51]. They consist in one-degree of freedom PID controllers designed to guarantee a target robustness (M_s) of 1.4 for servo purposes [51]:

$$u_x(s) = K_{PID} \cdot \left[\left(1 + \frac{1}{T_{i,PID} s} \right) \cdot e_x(s) + \frac{T_{d,PID} s}{\alpha \cdot T_{d,PID} s + 1} \cdot y_x(s) \right] \quad (14)$$

where $u_x(s)$ is the control signal, $y_x(s)$ the output signal of $P_x(s)$, and $e_x(s) = r_x(s) - y_x(s)$ the error between the reference ($r_x(s)$) and $y_x(s)$ signals. α, which is devoted to making proper the derivative part of the controller, equals to 0.1. The parameters of each conventional PID, i.e., their gain K_{PID}, their integral time $T_{i,PID}$ and their derivative time $T_{d,PID}$ can be observed in Table 2.

Thus, data generated by these PID structures are considered to determine the TSM values whenever a closed-loop configuration is used. The output of the target process being controlled by the conventional PID as well as the reference signal are then considered as the inputs of the ANN-based PID structure. The outputs of each ANN layer are later adopted in the TSM computation.

Results regarding TSM values when the default PIDs are considered are shown in Table 3. In this case, it is observed that TSM values show the same tendency as in the open-loop configuration. For instance, the transference of the ANN-based PID from $P_s(s)$ into $P_2(s)$ and into $P_3(s)$ can be considered since all TSM values are over the suitability limit, i.e., over 0.7. The minimum TSM value for $P_2(s)$ equals to 0.9472 and 0.9880 for the fix and variable set-points, respectively. These values change until 0.9499 for the fix set-point and 0.9872 for the variable one in case of being transferred into $P_3(s)$. These results clearly state that the ANN-based PID can be adopted as the ANN-based structure controlling either $P_2(s)$ or $P_3(s)$, regardless of the considered set-point. Nevertheless, the situation completely changes when $P_4(s)$ is considered. The minimum TSM value equals to 0.3926 and 0.5428 for the fix and variable set-points, respectively.

Regarding the time needed to compute the TSM, the same behaviour as in the Open-loop Data-generation scenario is observed. Now, the average time required for a variable set-point equals to 156.14 s whereas it diminishes until 34.25 s for a fix set-point. As stated before, these times are motivated by the complexity of the process being controlled as well as the topology of the set-point. Taking this into account, the objective of speeding-up the control design process and achieve the scalability of it can be fulfilled. The scalability is achieved since no extra controllers have to be designed nor trained. Moreover, in terms of the design acceleration, the times involved in the design of a new control structure are much higher than those required in transferring a controller [38]. Besides, TSM could be computed periodically and therefore, determine when the control structure needs to be updated (fine-tuned or retrained with new data) instead of deriving new solutions.

4.2.3. Discussion

TSM metric has been validated on two different configurations of the environment: (i) an open-loop and (ii) closed-loop data-generation configurations. In both cases, results show that the TSM is behaving as it should. Large values are provided for those situations where the controller can be transferred whereas low values are obtained otherwise. Nevertheless, results also show that there are some pros and cons that must be considered:

Table 3
TSM between correlations when transferring the $P_s(s)$ data-based controller to manage $P_2(s)$, $P_3(s)$ and $P_4(s)$ systems.

TSM from $P_s(s)$ to $P_2(s)$, $P_3(s)$ and $P_4(s)$ systems						
Layers	$P_s(s)$ to $P_2(s)$		$P_s(s)$ to $P_3(s)$		$P_s(s)$ to $P_4(s)$	
	Fix	Variable	Fix	Variable	Fix	Variable
Input	0.9867	0.9953	0.9789	0.9955	0.8067	0.9872
LSTM 1	0.9472	0.9902	0.9499	0.9895	0.8667	0.6667
LSTM 2	0.9477	0.9880	0.9504	0.9872	0.8298	0.6588
MLP 1	0.9476	0.9890	0.9504	0.9885	0.8019	0.6765
MLP 2	0.9473	0.9929	0.9500	0.9924	0.3926	0.5428
Minimum	0.9472	0.9880	0.9499	0.9872	0.3926	0.5428
Computation Time - (s)	34.32	153.28	34.41	161.09	34.02	154.06

- When TSM is applied over an open-loop data-generation scenario, the actuation of the controller cannot be considered. This implies that the loop should be closed to determine the ANN-based controller. This is performed considering the identification of the target processes, allowing us to obtain the required data in an easy way by using the process model.
- On the other hand, we can find that the target environments are being managed by a default controller, i.e., PI, PID and so on. In this case, the metric can be computed considering historical measurements from the conventional controller behaviour. The problem here lies in the fact that the actuation of the controller over the target environment is present in the data and therefore, this can mask the similarity between source and target domains. Nevertheless, the identification of the system is not required and therefore, the process of designing a controller can be speed-up.

In both cases, the pros we can obtain using the TSM metric are more beneficial. Moreover, the drawbacks can be considered as affordable since the deviations observed in the performance of the transferred controllers are really small.

It is worth noticing that one of the objectives of TSM is to achieve the speed-up of the control design process. This speeding is achieved by means of the TL process itself and the adoption of TSM. The speeding achieved with the TL is clear: the amount of time involved in the training of an ANN-based controller from scratch is much higher than the time involved in a probable fine-tuning process [38]. Moreover, the amount of time involved in the TSM computation must be considered. In both cases, it is observed that it depends on the topology of set-point as well as on the complexity of the process under control.

5. WWTP TSM application

5.1. WWTP environment

The industrial environment where the proposed metric is going to be applied corresponds to a WWTP, but instead of computing the TSM values over the real plant, the Benchmark Simulation Model No. 1 (BSM1) is considered [65]. It corresponds to a general purpose WWTP digital model implemented in Matlab® R2020b and Simulink® v.10.2. It consists of five reactor tanks connected in series. The first two, Tank 1 and Tank 2, correspond to anoxic tanks with a volume of 1000 m^3 each. Its main objective is to carry out the denitrification process where the amount of nitrates present in the residual urban waters are transformed intro nitrogen products [66]. Later, these products are managed in the last three reactor tanks, Tank 3, Tank 4, and Tank 5, which consist of aerated reactors with a volume of 1333 m^3 each. They are in charge of transforming the nitrogen products, especially the ammonia (S_{NH}) into nitrates, by means of the nitrification process [66].

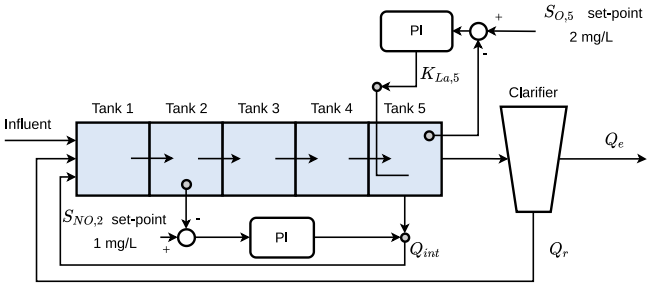


Fig. 11. BSM1 model with its default PI controllers. $K_{La,5}$, Q_{int}, Q_e and Q_r are the oxygen transfer coefficient, the internal recirculation flow, the effluent and the external recirculation flow, respectively.

Moreover, BSM1 also presents two default PI controllers managing the crucial WWTP control loops: (i) the dissolved oxygen (DO) and (ii) the nitrate–nitrogen (NO) [67]. Both of them are in charge of ensuring the correct performance of the nitrification and denitrification processes described by the Activated Sludge Model No. 1 (ASM1) [68]. In that sense, the DO control loop is in charge of maintaining the dissolved oxygen in the fifth reactor of the plant ($S_{O,5}$) at a fixed set-point of 2 mg/L. The NO one is in charge of maintaining the nitrate–nitrogen of the second reactor tank ($S_{NO,2}$) at a fixed value of 1 mg/L (see Fig. 11).

This is performed by means of manipulating the oxygen transfer coefficient at the fifth tank ($K_{La,5}$) and the internal recirculation flow rate (Q_{int}). In that sense, the control signals generated by the PI controllers consist in:

$$K_{La,5} = K_{S_{O,5}} \left[1 + \frac{1}{Ti_{S_{O,5}} s} \right] \cdot e_{S_{O,5}}(s) \tag{15}$$

$$Q_{int} = K_{Q_{int}} \left[1 + \frac{1}{Ti_{Q_{int}} s} \right] \cdot e_{S_{NO,2}}(s) \tag{16}$$

where $e_{S_{O,5}}(s)$ and $e_{S_{NO,2}}(s)$ are the errors between the desired set-points and the measured concentrations. $K_{S_{O,5}}$ is the PI proportional gain, which equals to 25, and $Ti_{S_{O,5}}$ is the PI integral time, which is set to $1 \cdot 10^{-3}$ days. In terms of the $S_{NO,2}$ PI controller, its gain, $K_{Q_{int}}$, equals to 10 000 whilst its integral time, $Ti_{Q_{int}}$, is equal to 0.0150 days. As it can be observed in the PI integral time, both control loops differ in their dynamics. The ones of $S_{O,5}$ are much faster than the dynamics of $S_{NO,2}$ concentrations. This is also corroborated by the BSM1 simulation protocol. It not only specifies how to perform a simulation, but also which topology of sensors and the sampling times that should be considered. In that sense, the sensors in charge of measuring the $S_{O,5}$ concentration return values every minute while the sensors of $S_{NO,2}$ return a value every 10 min [65,67]. Nevertheless, BSM1 also specifies that outputs from the benchmark should be obtained every 15 min. In other words, among all the available measurements, only those observed every 15 min must be considered [65].

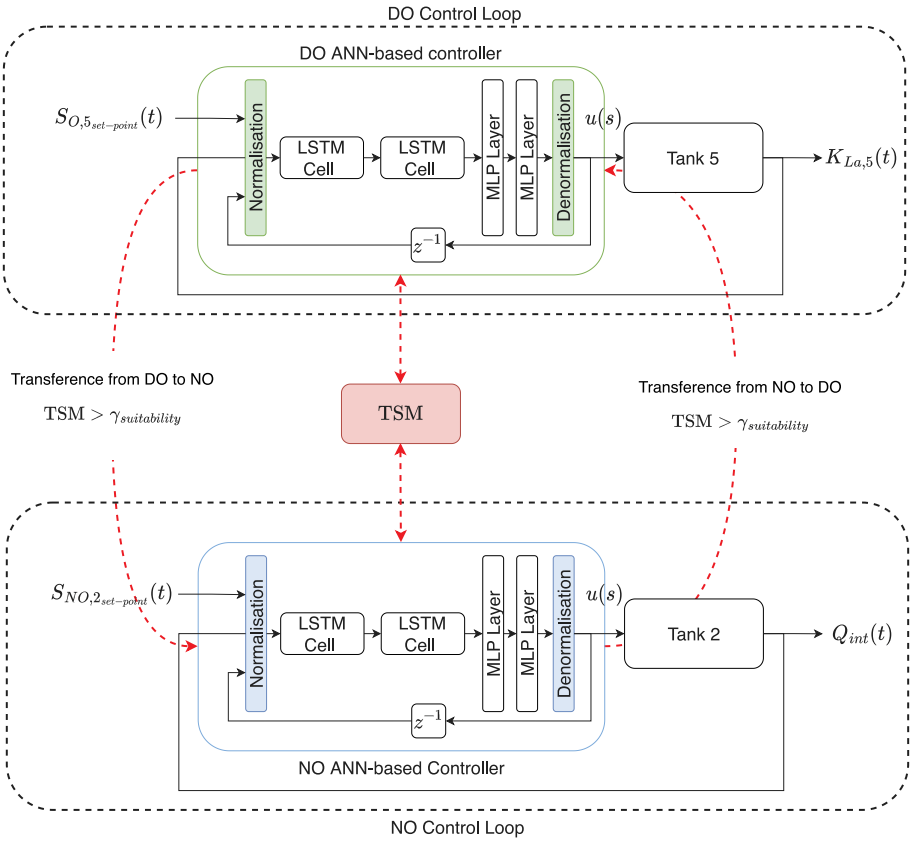


Fig. 12. Controllers designed to control the DO and NO control loops of a general purpose WWTP.

BSM1 also presents its own performance metrics. Since a WWTP environment is in charge of cleaning polluted waters, BSM1 implements the Effluent Quality Index (EQI) and the Overall Cost Index (OCI) metrics to determine the environmental performance and the costs related to the cleaning procedure. In terms of the control performance, BSM1 proposes the IAE and the ISE between the desired set-point and the obtained value as the metric measuring this performance [67]. After simulating an arbitrarily amount of time, they are computed as:

$$\mathrm{IAE} = \int_{t_0}^{t_f} |r_i(t) - y_i(t)|\, dt \qquad \mathrm{ISE} = \int_{t_0}^{t_f} (r_i(t) - y_i(t))^2\, dt \qquad (17)$$

where t_0 and t_f are the initial and final time of the performed simulation. $r_i(t)$ and $y_i(t)$ correspond to the desired set-point and the measured value of the concentration being controlled. If a whole year of the BSM1 behaviour is considered, for instance, the default PIs performance is as follows: (i) the $S_{O,5}$ IAE and ISE are equal to 12.1271 and 1.0378, and (ii) the IAE and ISE for the $S_{NO,2}$ equal to 71.9769 and 30.0537, respectively.

As it is observed, the performance of the $S_{NO,2}$ control loop can be improved. Different approaches and control structures could be tested, however, in this work we propose the TSM metric in order to transfer a unique ANN-based controller performing well between the WWTP control loops, i.e., the DO and NO. In that manner, this will also ease the design process of the WWTP operation.

5.2. Results

After the validation of the TSM metric, it is applied over the WWTP environment, especially, over the BSM1 [65]. Thus, the main idea is to develop and implement an ANN-based PI for the DO BSM1 control loop and then transfer it into the NO or vice-versa. For that reason, two ANN-based PIs are considered. Their structure is adopted from the work presented in [38] and corresponds to the structure shown in Fig. 12.

The ANN-based PI structures are trained again considering TensorFlow in its 1.14 version over Python 3.6 and the ADAM optimiser with back-propagation training algorithm. The learning rate is initially set at $1 \cdot 10^{-3}$ and the L_2 penalty at $1 \cdot 10^{-4}$. Once trained, the controller is deployed over Matlab® R2020b and Simulink® v.10.2. The prediction performance for the ANN-based PI managing the DO control loop returns a RMSE, MAPE and R^2 equal to 0.0260, 1.3561% and 0.999, respectively. The structure in charge of the NO control loop shows a RMSE equal to 0.0444, a MAPE to 5.2018% and a R^2 to 0.997. In both cases, these metrics are computed contrasting the measurements provided by the ANN-based PIs with the ones provided by the default BSM1 controllers.

In terms of their control performance, it is computed substituting the default PIs by the ANN-based PIs in the digital BSM1 model. It is worth noting here that no transfer is performed yet. As a result, the IAE and ISE metrics for each BSM1 control loop are equal to: (i) 0.5104 and 0.0014, respectively, for the DO control loop, and (ii) 53.0984 and 16.6730, for the NO one. When default BSM1 PI controllers are considered instead, the IAE and ISE for the DO equal to 12.1271 and 1.0378. For the NO, these values equal to 71.9769 and 30.0537, respectively. Thereby, it is shown that both ANN-based structures can show a good control performance when they are trained for managing their respective control loops (see Fig. 13).

Once the control performance is computed and the behaviour of both ANN-based controllers is determined, the TSM metric is computed considering the closed loop configuration, i.e., the historic data from the target domain. Results are shown in Table 4.

At first sight it is observed that in all cases the transference of the ANN-based controller is suitable whatever direction is chosen.

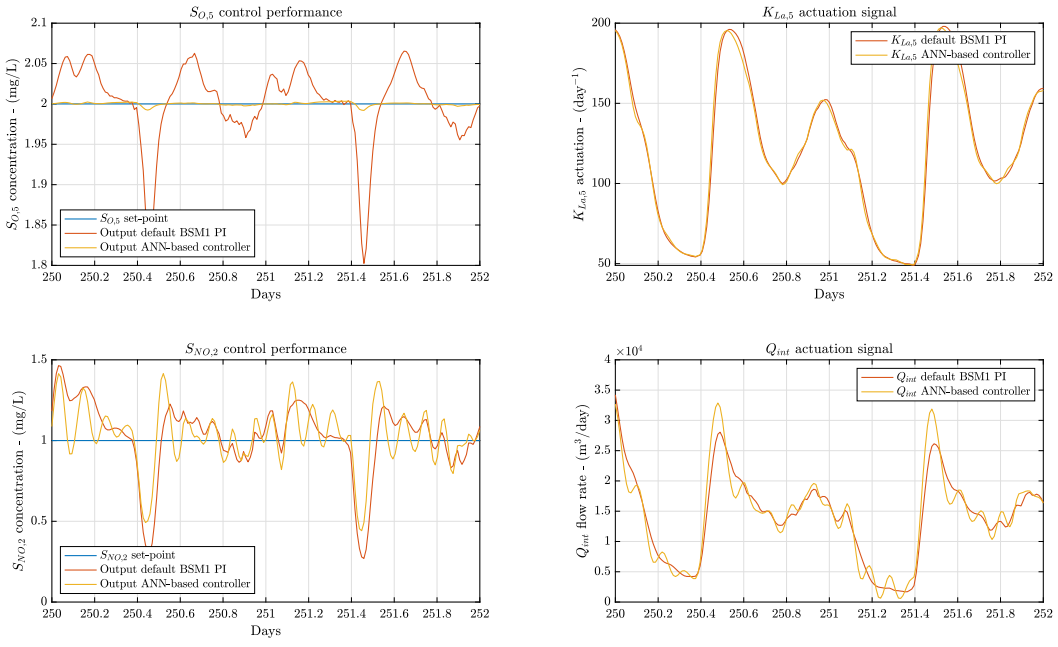


Fig. 13. Control Performance of the BSM1 scenario when the dedicated ANN-based controllers are considered.

Table 4
TSM when transferring the BSM1 data-based controllers from one control loop to another.

TSM between BSM1 control loops			
Layers	From $S_{O,5}$ to $S_{NO,2}$	From $S_{NO,2}$ to $S_{O,5}$	Without transfer
Input	0.8930	0.7320	0.8262
LSTM 1	0.9274	0.7718	0.9324
LSTM 2	0.9144	0.8910	0.9018
MLP 1	0.9148	0.8910	0.9078
MLP 2	0.9475	0.9246	0.9518
Minimum	0.8930	0.7320	0.8262
Computation Time - (s)	158.23	162.63	173.63

For instance, the minimum TSM regarding the transference from the DO control loop to the NO one is equal to 0.8930. When the other direction is chosen, the TSM value is reduced until 0.7320. This entails that the performance behaviour of the transferred version of the ANN-based controller is not as good as the one primarily obtained. Thus, it is corroborated that the more similar the scenarios, the higher the transferability between ANN-based controllers.

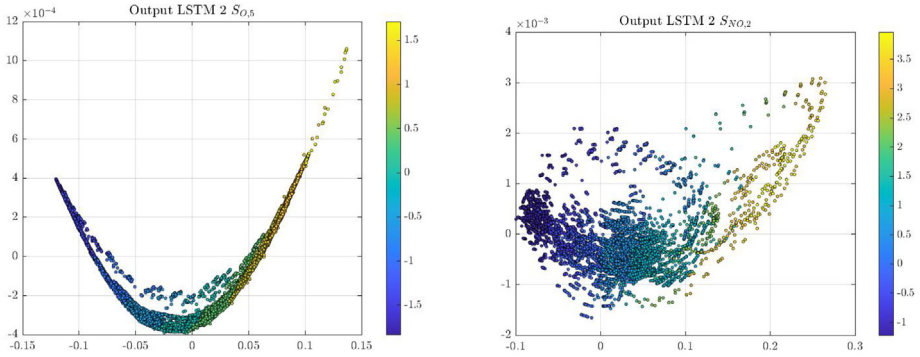
If the values shown in Table 4 are considered, it is clearly observed that the best option is to develop an ANN-based controller for the DO control loop and then transfer it to the NO one. If this is performed, all the TSM values between correlation matrices are really close or even higher than 0.9, which tells us that transferring this controller is a very suitable option. The highest similarity is now observed in the output of the MLP 2 layer (see Fig. 14). As it is observed, the same tendency in the values is presented and therefore, the same shape (convex curve) is obtained as the temporal evolution of the outputs of the ANN-based controller's layer. This means that at this point is where the ANN-based controller considering measurements from the NO control loop is giving values much closer to the ones observed when considering data from the DO loop.

Regarding the time involved in the TSM computation, again one can observe that in average it equals to 164.83 s. In this case, the set-point is a fixed value, but the process under control is much more complicated than the academic examples considered before. Nevertheless, these amounts of time are small enough to determine that adopting the TSM metric allows the control designer to not only obtain a new control structure from an existing one, but also to speed-up the whole process.

In terms of the IAE and ISE values, the transference of the ANN-based PI from the DO control loop to the NO produce a reduction with respect to the IAE and ISE values given by the default BSM1 PI controllers. In this case, the IAE and ISE are reduced from 12.1271 and 1.0378 to 0.5104 and 0.0014, respectively, for the DO control loop. In terms of the NO loop, these values are reduced from 71.9769 and 30.0537 to 42.2648 and 11.6644, respectively.

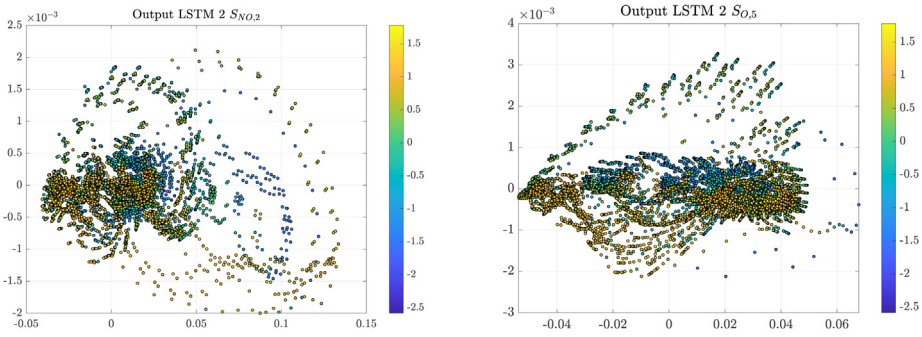
The same principle is observed when transferring the ANN-based controller from the NO control loop to the DO one. Here, TSM values between the autocorrelation and cross-correlation matrices decrease motivated by the fact that now, the outputs of the different layers conforming the ANN-based controllers, are less similar than before (see Fig. 15). Even though TSM tells that the ANN-based structure can be transferred from the DO to NO control loop, the control performance can experience some degradation (see Table 5). This is corroborated by the IAE and ISE values for both control loops. Experiencing improvements of 15.37% and 67.71%, the IAE and ISE values for the DO control loop equal to 10.2629 and 0.3351, respectively. In terms of the NO control loop, these values equal to 53.0984 and 16.6730, respectively.

As it has been previously observed, the control performance of the transferred ANN-based controller is highly related to the TSM value between domains. The higher the TSM, the better the control behaviour. For that reason, the best option to derive the WWTP control loops is to firstly train the ANN-based controller



(a) MDS LSTM 2 $S_{O,5}$ (b) MDS LSTM 2 $S_{NO,2}$

Fig. 14. Output of the MDS algorithm when dimensions of the different layers are reduced into a 2-dimensional space.



(a) MDS LSTM 2 $S_{NO,2}$ (b) MDS LSTM 2 $S_{O,5}$

Fig. 15. Output of the LSTM 2 layer when the MDS algorithm is applied over the ANN-based controller managing $S_{NO,2}$ and then transferred into $S_{O,5}$ control loop.

Table 5
Control performance when ANN-based PI controllers are transferred between BSM1 control loops.

Control performance				
	DO control loop		NO control loop	
	IAE	ISE	IAE	ISE
Default BSM1 PI controllers	12.1271	1.0378	71.9769	30.0537
TL from DO to NO	0.5104	0.0014	42.2648	11.6644
TL from NO to DO	10.2629	0.3351	53.0984	16.6730

to correctly manage the DO control loop and then transfer it into the NO. However, the performance yielded by this ANN-based controller is far from achieving its main objective. As it is observed in Fig. 16, the tracking process of the $S_{NO,2}$ concentration is not fulfilled.

To alleviate this, a fine-tuning process can be performed so as to adapt the behaviour of the ANN-based controller to the target domain and therefore, achieve the desired objective [38]. For instance, the performance of the NO control loop is improved until showing IAE and ISE values equal to 7.0930 and 0.4528, respectively. This entails that the fine-tuning of the ANN-based controller in the target domain can improve the default control performance a 90.15% and a 98.49%, respectively. In such a context, the final control performance of the ANN-based structures can be observed in Fig. 17.

To summarise, the TSM tells if an ANN-based structure can be transferred from a source environment to a target one offering a reliable performance. This entails a great benefit for the control designer since instead of focusing the efforts in designing as many controllers as loops, the designer only needs to design a controller that can be transferred into the different control loops. In that process, the TSM is of the utmost utility since it tells the suitability of transferring an ANN structure beforehand.

6. Conclusions

This work proposes a solution for a critical issue regarding the adoption of Transfer Learning methods to derive new industrial controllers. The transference of ANN-based controllers cannot be freely performed since the performance of the ANN-based controller in the target control loop cannot be known *a priori*. For that reason, the efforts of this work are focused on developing a new metric, the TSM. It bases its behaviour on the computation of the similarity between the source and target domains. To achieve this, the computation of correlations between the outputs of the different layers of the ANN-based controller when it is dealing with measurements from the source and target domains is proposed. Thus, the principle behind the TSM can be summarised as follows: the more similar the correlation matrices, the more suitable the transfer.

Here, the validation of the TSM, and therefore its behaviour, is tested considering four different controlled processes, two FOPDT and two SOPDT processes. Results show that:

- TSM yields values close to the maximum transfer suitability (maximum value equals to 1) between those environments or processes where the ANN-based controller performs well for the source and target domain.

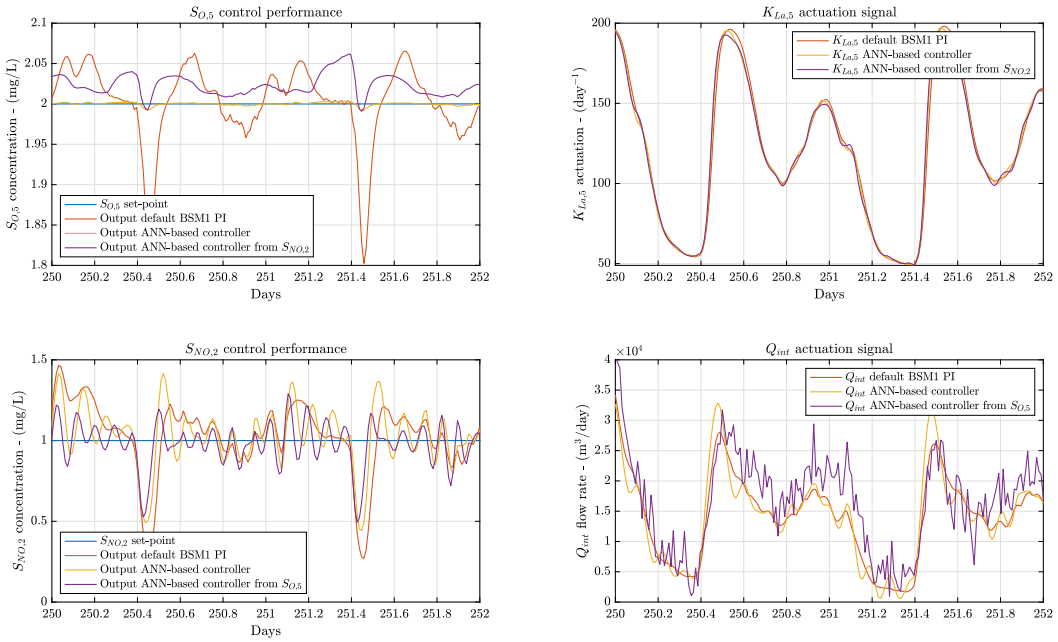


Fig. 16. Control Performance of the BSM1 scenario when the ANN-based controllers are transferred.

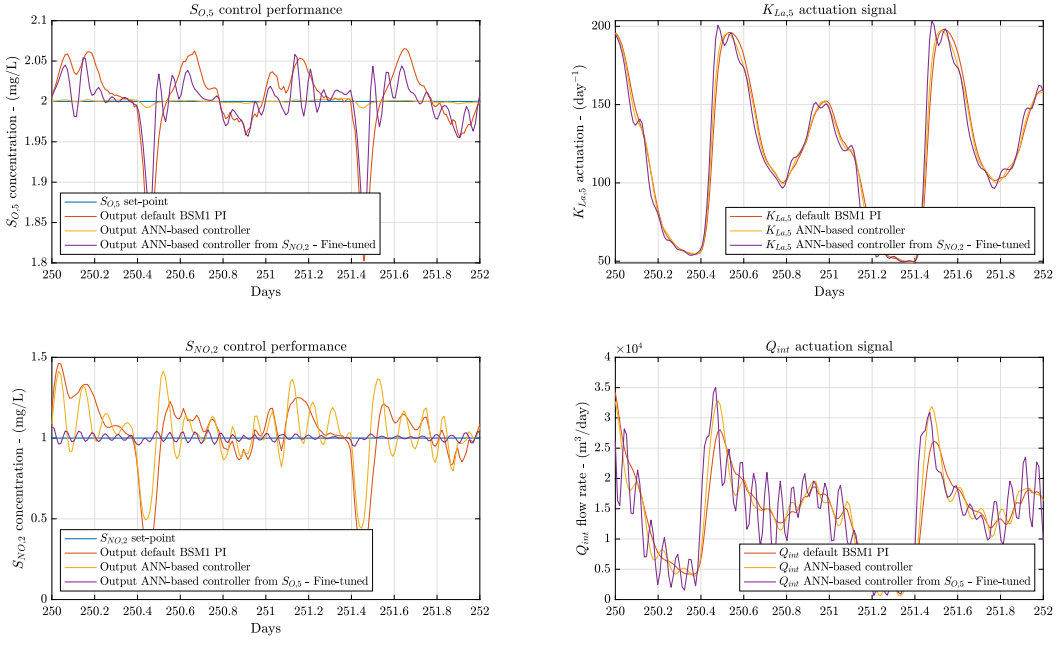


Fig. 17. Control Performance of the BSM1 scenario when the ANN-based controllers are transferred and fine-tuned.

- TSM returns values far from the maximum suitability in those scenarios where the transference of the ANN-based controller is not feasible at all.

Once the TSM behaviour is validated, it is applied over a critical industrial environment, a WWTP, so as to determine if the transference of an ANN-based controller can be performed without problems. Results show that:

- ANN-based structures substituting the default controllers can be applied and transferred regardless the direction, i.e., from the DO to the NO control loop and vice-versa.
- The highest transferability is the one concerning the transference from the DO to the NO control loop. The minimum TSM value equals to 0.8930.
- In terms of the control performance, the highest improvement in the IAE and ISE values is offered when the ANN-based controller is transferred from the DO to the NO control loops. IAE and ISE values are reduced in average a 68.54% and an 80.53%, respectively.

All this motivates us to conclude that TSM can be of utmost interest to decide when to apply TL techniques and if it is suitable to consider ANN-based controllers instead of conventional ones.

Moreover, this can pave the way in the application of ANN-based controllers and TL techniques in the industrial control domain, especially, in the design process.

Abbreviations

The following abbreviations and notation are used in this manuscript:

α	Parameter to make proper the derivative part of the controller
ANN	Artificial Neural Network
ASM1	Activated Sludge Model N.1
BSM1	Benchmark Simulation Model No. 1
CNN	Convolutional Neural Network
D	Number of dimensions of an ANN layer
DO	Dissolved Oxygen control loop
EQI	Effluent Quality Index
$e_{S_{O,5}}$	Error between the desired $S_{O,5}$ set-point and the measured concentration
$e_{S_{NO,2}}$	Error between the desired $S_{NO,2}$ set-point and the measured concentration
FFNN	Feed-forward Neural Network
FOPDT	First Order Plus Dead-time
FT	Fine-tuning
IAE	Integrated Absolute Error
ICA	Independent Component Analysis
IMC	Internal Model Controller
ISE	Integrated Squared Error
K	Gain of the process
$K_{La,5}$	Oxygen Transfer Coefficient of the 5th reactor tank
K_{PID}	Proportional action of the PID controller
L	Delay of the process
L_2	L2 Norm
LDA	Linear Discriminant Analysis
LSTM	Long Short–Term Memory
M	Number of layers of the ANN-based controller
MAPE	Mean Average Percentage Error
MDS	Multidimensional Scaling Algorithm
MLP	Multilayer Perceptron
MPC	Model Predictive Controller
N	Number of considered time instants
NO	Nitrate and nitrite nitrogen control loop
OCI	Overall Cost Index
$P(s)$	Process under control
$P_{i,x}$	Identified process of $P_x(s)$
$P_s(s)$	Source domain process
$P_t(s)$	Target domain process
PCA	Principal Components Analysis
PI	Proportional Integral controller
PID	Proportional Integral Derivative controller
Q_{int}	Internal recirculation flow rate
$r(s)$	Reference signal
$\mathbf{r}_{x'^i_j}$	Sample autocorrelation vector for $\left[\mathbf{X}'^{,i}\right]$
$\mathbf{r}_{x'y'^i_j}$	Sample cross-correlation vector for $\left[\mathbf{Y}'^{,i}\right]$
R^2	Determination coefficient
$\left[\mathbf{R}_{x',i}\right]_{k,j}$	Element from k row and j column of $\mathbf{R}_{x',i}$
$\left[\mathbf{R}_{x'y',i}\right]_{k,j}$	Element from k row and j column of $\mathbf{R}_{x'y',i}$
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
$S_{NO,2}$	Nitrate and nitrite nitrogen in the 2th reactor tank
S_{NH}	Ammonium concentration
$S_{O,5}$	Dissolved oxygen concentration in the 5th reactor tank
SOPDT	Second Order Plus Dead-time
T	Time constant of the process
$T_{d,PID}$	Derivative action of the PID controller
t_f	Final time of the performed simulation
$T_{i,PID}$	Integral action of the PID controller
TL	Transfer Learning
TSM	Transfer Suitability Metric
t_O	Initial time of the performed simulation
$u(s)$	Actuation signal
ω_n	Natural frequency of the process
WWTP	Wastewater Treatment Plant
$\mathbf{x}^i$	Output of the ith layer of the ANN-based controller form the source domain at time instant t
$\mathbf{X}^i$	Output of the ith layer of the ANN-based controller form the source domain
$\mathbf{X}'^{,i}$	$\mathbf{X}^i$ after the application of the MDS algorithm
$\left[\mathbf{X}'^{,i}\right]_{.,j}$	Column j of $\mathbf{X}'^{,i}$
$\mathbf{X}^{\Delta,i}_j$	Auxiliary matrix to compute the correlations
$\mathbf{y}^i$	Output of the ith layer of the ANN-based controller form the target domain at time instant t
$\mathbf{Y}^i$	Output of the ith layer of the ANN-based controller form the target domain
$\mathbf{Y}'^{,i}$	$\mathbf{Y}^i$ after the application of the MDS algorithm
$\left[\mathbf{Y}'^{,i}\right]_{.,j}$	Column j of $\mathbf{Y}'^{,i}$
$y(s)$	Measured output signal
ζ	Damping coefficient
$\gamma_{suitability}$	Transfer Suitability limit

CRediT authorship contribution statement

Ivan Pisa: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft, Visualization, Writing – review & editing. **Antoni Morell:** Investigation, Writing – review & editing, Supervision, Project administration, Funding acquisition. **Jose Lopez Vicario:** Investigation, Writing – review & editing, Supervision, Project administration, Funding acquisition. **Ramon Vilanova:** Investigation, Writing – review & editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

This work has received support from the Catalan Government under Project 2021 SGR 00197, and also by the Spanish Government under MICIN project PID2019-105434RB-C33 co-funded with the European Union ERDF funds and under MCIN/AEI/10.13039/501100011033 project TED2021-129134B-I00 co-funded with the European Union "NextGenerationEU"/PRTR funds.

References

[1] A. Ustundag, E. Cevikcan, Industry 4.0: Managing the Digital Transformation, Springer, Cham, Switzerland, 2017.
[2] M. Wollschlaeger, T. Sauter, J. Jasperneite, The future of industrial communication: Automation networks in the era of the internet of things and industry 4.0, IEEE Ind. Electron. Mag. 11 (2017).

[3] A.A. Cook, G. Mısırlı, Z. Fan, Anomaly detection for IoT time-series data: A survey, IEEE Internet Things J. 7 (2019) 6481–6494.
[4] X. Wang, H. Liu, Soft sensor based on stacked auto-encoder deep neural network for air preheater rotor deformation prediction, Adv. Eng. Inform. 36 (2018) 112–119.
[5] S. Khatir, S. Tiachacht, C.L. Thanh, E. Ghandourah, S. Mirjalili, M.A. Wahab, An improved artificial neural network using arithmetic optimization algorithm for damage assessment in FGM composite plates, Compos. Struct. 273 (2021) 114287.
[6] L.V. Ho, D.H. Nguyen, M. Mousavi, G.D. Roeck, T. Bui-Tien, A.H. Gandomi, M.A. Wahab, A hybrid computational intelligence approach for structural damage detection using marine predator algorithm and feedforward neural networks, Comput. Struct. 252 (2021) 106568.
[7] L.V. Ho, T.T. Trinh, G.D. Roeck, T. Bui-Tien, L. Nguyen-Ngoc, M.A. Wahab, An efficient stochastic-based coupled model for damage identification in plate structures, Eng. Fail. Anal. 131 (2022) 105866.
[8] B. Cai, K. Hao, Z. Wang, C. Yang, X. Kong, Z. Liu, R. Ji, Y. Liu, Data-driven early fault diagnostic methodology of permanent magnet synchronous motor, Expert Syst. Appl. 177 (2021).
[9] X. Kong, B. Cai, Y. Liu, H. Zhu, C. Yang, C. Gao, Y. Liu, Z. Liu, R. Ji, Fault diagnosis methodology of redundant closed-loop feedback control systems: Subsea blowout preventer system as a case study, IEEE Trans. Syst. Man Cybern.: Syst. (2022).
[10] B. Cai, H. Fan, X. Shao, Y. Liu, G. Liu, Z. Liu, R. Ji, Remaining useful life re-prediction methodology based on Wiener process: Subsea christmas tree system as a case study, Comput. Ind. Eng. 151 (2021).
[11] F.A. Souza, R. Araújo, J. Mendes, Review of soft sensor methods for regression applications, Chemometr. Intell. Lab. Syst. 152 (2016) 69–79.
[12] S. Wang, H. Wang, Y. Zhou, J. Liu, P. Dai, X. Du, M.A. Wahab, Automatic laser profile recognition and fast tracking for structured light measurement using deep learning and template matching, Measurement 169 (2021) 108362.
[13] A. Rani, V. Singh, J. Gupta, Development of soft sensor for neural network based control of distillation column, ISA Trans. 52 (2013) 438–449.
[14] I. Pisa, I. Santín, J.L. Vicario, A. Morell, R. Vilanova, ANN-based soft sensor to predict effluent violations in wastewater treatment plants, Sensors 19 (2019) 1280.
[15] V.d.J. da Silva Ribeiro, G.F. de Moraes Oliveira, M. Cristian, A.L. Martins, L.D. Fernandes, M.P. Vega, Neural network based controllers for the oil well drilling process, J. Pet. Sci. Eng. 176 (2019) 573–583.
[16] N.K. Kandasamy, G. Karunagaran, C. Spanos, K.J. Tseng, B.-H. Soong, Smart lighting system using ANN-IMC for personalized lighting control and daylight harvesting, Build. Environ. 139 (2018) 170–180.
[17] W. Dong, S. Li, X. Fu, Z. Li, M. Fairbank, Y. Gao, Control of a buck DC/DC converter using approximate dynamic programming and artificial neural networks, IEEE Trans. Circuits Syst. I. Regul. Pap. 68 (4) (2021) 1760–1768.
[18] D. Wang, Z.J. Shen, X. Yin, S. Tang, X. Liu, C. Zhang, J. Wang, J. Rodriguez, M. Norambuena, Model predictive control using artificial neural network for power converters, IEEE Trans. Ind. Electron. 69 (4) (2021) 3689–3699.
[19] I. Santín, C. Pedret, R. Vilanova, M. Meneses, Advanced decision control system for effluent violations removal in wastewater treatment plants, Control Eng. Pract. 49 (2016).
[20] C. Foscoliano, S. Del Vigo, M. Mulas, S. Tronci, Predictive control of an activated sludge process for long term operation, Chem. Eng. J. 304 (2016) 1031–1044.
[21] J.-F. Qiao, Y. Hou, L. Zhang, H.-G. Han, Adaptive fuzzy neural network control of wastewater treatment process with multiobjective operation, Neurocomputing 275 (2018) 383–393.
[22] I. Pisa, A. Morell, J.L. Vicario, R. Vilanova, Denoising autoencoders and LSTM-based artificial neural networks data processing for its application to internal model control in industrial environments—The wastewater treatment plant control case, Sensors 20 (2020) 3743.
[23] M.A. Amirabadi, M.H. Kahaei, S.A. Nezamalhosseini, Novel suboptimal approaches for hyperparameter tuning of deep neural network [under the shelf of optical communication], Phys. Commun. 41 (2020) 101057.
[24] G.G. Wang, M. Lu, Y.Q. Dong, X.J. Zhao, Self-adaptive extreme learning machine, Neural Comput. Appl. 27 (2016) 291–303.
[25] J.H. Yi, J. Wang, G.G. Wang, Improved probabilistic neural networks with self-adaptive strategies for transformer fault diagnosis problem, Adv. Mech. Eng. 8 (2016) 1–13.
[26] Y. Wang, X. Qiao, G.G. Wang, Architecture evolution of convolutional neural network using monarch butterfly optimization, J. Ambient Intell. Humaniz. Comput. 1 (2022) 1–15.
[27] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, Q. He, A comprehensive survey on transfer learning, Proc. IEEE 109 (2020) 43–76.
[28] T. Han, C. Liu, W. Yang, D. Jiang, Deep transfer network with joint distribution adaptation: A new intelligent fault diagnosis framework for industry application, ISA Trans. 97 (2020) 269–281.
[29] S. Shao, S. McAleer, R. Yan, P. Baldi, Highly accurate machine fault diagnosis using deep transfer learning, IEEE Trans. Ind. Inform. 15 (2018) 2446–2455.
[30] T. Lu, F. Yu, C. Xue, B. Han, Identification, classification, and quantification of three physical mechanisms in oil-in-water emulsions using AlexNet with transfer learning, J. Food Eng. 288 (2021) 110220.
[31] F. Curreri, L. Patanè, M.G. Xibilia, RNN-and LSTM-based soft sensors transferability for an industrial process, Sensors 21 (2021) 823.
[32] P. Oliveira, B. Fernandes, C. Analide, P. Novais, Forecasting energy consumption of wastewater treatment plants with a transfer learning approach for sustainable cities, Electronics 10 (2021) 1149.
[33] Z. Ke, Z. Li, Z. Cao, P. Liu, Enhancing transferability of deep reinforcement learning-based variable speed limit control using transfer learning, IEEE Trans. Intell. Transp. Syst. 22 (7) (2020) 4684–4695.
[34] S. Xu, Y. Wang, Y. Wang, Z. O'Neill, Q. Zhu, One for many: Transfer learning for building hvac control, in: Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, 2020, pp. 230–239.
[35] Y. Tao, J. Qiu, S. Lai, A hybrid cloud and edge control strategy for demand responses using deep reinforcement learning and transfer learning, IEEE Trans. Cloud Comput. 10 (1) (2021) 56–71.
[36] P. Lissa, M. Schukat, M. Keane, E. Barrett, Transfer learning applied to DRL-based heat pump control to leverage microgrid energy efficiency, Smart Energy 3 (2021).
[37] I. Pisa, A. Morell, J.L. Vicario, R. Vilanova, Transfer learning approach for the design of basic control loops in wastewater treatment plants, in: 2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), IEEE, 2021, pp. 1–8.
[38] I. Pisa, A. Morell, R. Vilanova, J.L. Vicario, Transfer learning in wastewater treatment plant control design: From conventional to long short-term memory-based controllers, Sensors 21 (2021) 6315.
[39] Z. Wang, Z. Dai, B. Póczos, J. Carbonell, Characterizing and avoiding negative transfer, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 11293–11302.
[40] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, Cambridge, MA, USA, MIT Press, 2016.
[41] J. Yosinski, J. Clune, Y. Bengio, H. Lipson, How transferable are features in deep neural networks? Adv. Neural Inf. Process. Syst. 27 (2014).
[42] J. Williams, A. Tadesse, T. Sam, H. Sun, G.D. Montanez, Limits of transfer learning, in: International Conference on Machine Learning, Optimization, and Data Science, Springer, 2020, pp. 382–393.
[43] M.J. Afridi, A. Ross, E.M. Shapiro, On automated source selection for transfer learning in convolutional neural networks, Pattern Recognit. 73 (2018) 65–75.
[44] C. Klemenjak, A. Faustine, S. Makonin, W. Elmenreich, On metrics to assess the transferability of machine learning models in non-intrusive load monitoring, 2019, arXiv preprint arXiv:1912.06200.
[45] S. Boudabous, S. Clémençon, H. Labiod, J. Garbiso, Dynamic graph convolutional LSTM application for traffic flow estimation from error-prone measurements: results and transferability analysis, in: 2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA), IEEE, 2021, pp. 1–10.
[46] R. Souissi, A. Al Bitar, M. Zribi, Accuracy and transferability of artificial neural networks in predicting in situ root-zone soil moisture for various regions across the globe, Water 12 (2020) 3109.
[47] A. Schilling, A. Maier, R. Gerum, C. Metzner, P. Krauss, Quantifying the separability of data classes in neural networks, Neural Netw. 139 (2021) 278–293.
[48] I. Pisa, A. Morell, J.L. Vicario, R. Vilanova, Transfer learning suitability metric for ANN-based industrial controllers, in: 2022 27th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), IEEE, 2022, pp. 1–8.
[49] R. Vilanova, A. Visioli, PID Control in the Third Millennium - Lessons Learned and News Approaches, Springer, 2012.
[50] N. Lanzetti, Y.Z. Lian, A. Cortinovis, L. Dominguez, M. Mercangöz, C. Jones, Recurrent neural network based MPC for process industries, in: 2019 18th European Control Conference (ECC), IEEE, 2019, pp. 1005–1010.
[51] R. Kurokawa, T. Sato, R. Vilanova, Y. Konishi, Discrete-time first-order plus dead-time model-reference trade-off PID control design, Appl. Sci. 9 (2019) 3220.
[52] R. Kurokawa, T. Sato, R. Vilanova, Y. Konishi, Design of optimal PID control with a sensitivity function for resonance phenomenon-involved second-order plus dead-time system, J. Franklin Inst. B 357 (2020) 4187–4211.
[53] J.D. Carroll, P. Arabie, Multidimensional scaling, Meas., Judgm. Decis. Mak. (1998) 179–250.
[54] M.C. Hout, M.H. Papesh, S.D. Goldinger, Multidimensional scaling, Wiley Interdiscip. Rev.: Cogn. Sci. 4 (2013) 93–103.
[55] G. Boquet, A. Morell, J. Serrano, J.L. Vicario, A variational autoencoder solution for road traffic forecasting systems: Missing data imputation, dimension reduction, model selection and anomaly detection, Transp. Res. C: Emerg. Technol. 115 (2020) 102622.
[56] B. Ratner, The correlation coefficient: Its values range between+ 1/- 1, or do they? J. Targeting Meas. Anal. Mark. 17 (2) (2009) 139–142.

[57] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G.S. Corrado, A. Davis, J. Dean, M. Devin, et al., TensorFlow: Large-scale machine learning on heterogeneous systems, 2015, Available online: https://www.tensorflow.org/ (accessed on October 2020).
[58] W. McKinney, et al., Data structures for statistical computing in python, in: Proceedings of the 9th Python in Science Conference, Vol. 445, 2010, pp. 51–56.
[59] T.E. Oliphant, A Guide to NumPy, Vol. 1, Trelgol Publishing, Spanish Fork, UT, USA, 2006.
[60] J.D. Hunter, Matplotlib: A 2D graphics environment, Comput. Sci. Eng. 9 (2007) 90–95.
[61] F. Pedregosa, et al., Scikit-learn: Machine learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830.
[62] P. Virtanen, R. Gommers, T.E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, et al., SciPy 1.0: Fundamental algorithms for scientific computing in python, Nature Methods 17 (2020) 261–272.
[63] D.S. Manu, A.K. Thalla, Artificial intelligence models for predicting the performance of biological wastewater treatment plant in the removal of Kjeldahl Nitrogen from wastewater, Appl. Water Sci. 7 (2017) 3783–3791.
[64] V.M. Alfaro, R. Vilanova, Control of high-order processes: repeated-pole plus dead-time models' identification, Int. J. Control (2021) 1–11.
[65] J.B. Copp, The Cost Simulation Benchmark: Description and Simulator Manual (Cost Action 624 and Action 682), Office for Official Publications of the European Union, Luxembourg, 2002.
[66] B. Halling-Sørensen, S.E. Jorgensen, The Removal of Nitrogen Compounds from Wastewater, Elsevier, Amsterdam, The Netherlands, 1993.
[67] J. Alex, L. Benedetti, J. Copp, K.-V. Gernaey, U. Jeppsson, I. Nopens, M.-N. Pons, L. Rieger, C. Rosen, J.-P. Steyer, P. Vanrolleghem, Benchmark Simulation Model No. 1 (BSM1), Tech. Rep., Department of Industrial Electrical Engineering and Automation, Lund University, 2008.
[68] M. Henze, W. Gujer, T. Mino, M.C.M. van Loosdrecht, Activated sludge models ASM1, ASM2, ASM2d and ASM3, IWA Publishing, London, UK, 2000.