Data-driven simulation for pedestrian avoiding a fixed obstacle

Rafael F. Martin[1*] and Daniel R. Parisi[1,2]

[1] Instituto Tecnológico de Buenos Aires. Lavarden 315, (C1437FBG) C. A. de Buenos Aires, Argentina.
*ramartin@itba.edu.ar
[2] Consejo Nacional de Investigaciones Científícas y Técnicas, Argentina.

Abstract. Data-driven simulation of pedestrian dynamics is an incipient and promising approach for building reliable microscopic pedestrian models. We propose a methodology based on generalized regression neural networks, which does not have to deal with a huge number of free parameters as in the case of multilayer neural networks. Although the method is general, we focus on the one pedestrian - one obstacle problem. The proposed model allows us to simulate the trajectory of a pedestrian avoiding an obstacle from any direction.

Keywords: pedestrian dynamics, data-driven simulation, navigation, steering, generalized regression neural network, artificial intelligence.

1 Introduction

Recently, we proposed a general framework of pedestrian simulation [1] in which the surroundings of a virtual pedestrian, i.e., obstacles and other noncontacting particles, can only influence its trajectory by modifying its desired velocity.

The basic assumption is that the avoidance behavior can be exerted only by the self-propelled mechanism of the particle itself (usually modeled by the desired velocity).

Under this approach, the problem lies in postulating the heuristics required for computing the variable desired velocity depending on the environment. As in traditional pedestrian theoretical models, any arbitrary heuristic can be proposed (for example, [2], [3]) and then the free parameters could be tuned in order to obtain simulated trajectories that approach experimental micro or macroscopic data.

Instead of this traditional methodology, we can directly use the experimental data so as to compute the desired velocity at each time step. From a set of real trajectories we extract the information for providing a desired velocity to the simulated agent, considering the state of the agent in the simulated and experimental environment.

Here we propose a data-driven approach using a nonparametric universal interpolator: the generalized regression neural network (GRNN) [4]. The GRNN needs to have access to the data examples (patterns) when predicting a new output. However, because it has only one degree of freedom (only one free parameter), the number of (input/output) patterns can be relatively low. Also, we postulate that a complete set of (input/output) examples, extracted from experimental trajectories, could be sufficient for simulating and reproducing several configurations. As a starting point, here we present this methodology in the case of one pedestrian avoiding a fixed obstacle.

2 The data-driven model

The set of experimental trajectories Because this is a data-driven model, the experimental data are the first ingredient needed. As a case study of the proposed method, we will focus on a simple configuration, considering one pedestrian and one fixed obstacle.

We realize several experiments to obtain real trajectories. Volunteers were instructed to walk from a starting points to a final point. Some of these trajectories have an obstacle in the way.

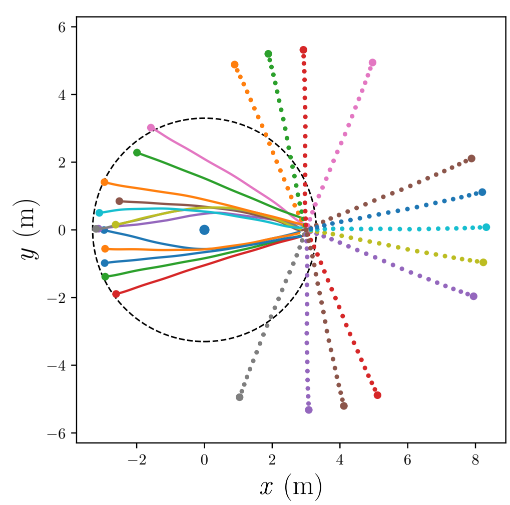


Fig. 1: Experimental trajectories (solid lines) and rotated trajectories (dashed lines). The points indicate the initial position of trajectories.

We choose 13 experimental trajectories represented in solid lines in Fig. 1. Another 13 trajectories were obtained by replicating and rotating extreme trajectories, which do not avoid the obstacle. The 26 trajectories will provide a set of data examples for solving the one pedestrian - one narrow obstacle problem after the following processing.

Input and output We postulate continuous input/state $\boldsymbol{\xi}_{ij}$ and output/reaction $\boldsymbol{\zeta}_i$ vectors given by

$$\boldsymbol{\xi}_{ij} = [\ |\hat{\mathbf{v}}|_i,\ \hat{\theta}_{ij},\ \hat{d_{ij}}\ ,\ \hat{\theta}^v_{ij},\ |\hat{\mathbf{v}}|_{ij},\ \hat{d}_{iT}] \tag{1}$$

$$\boldsymbol{\zeta}_i = [\mathbf{v}^+_i,\ \theta^+_i\ ] \tag{2}$$

The variables with hat are dimensionless versions of those presented in Fig. 2 (a), where the two velocity variables were divided by 1.8 m/s, the two distance variables were divided by 4 m saturating its values at 2, $\hat{\theta}_{ij}$ were divided by $\pi/2$ and saturates at -1 and 1 and

$$\hat{\theta}^v_{ij} = \begin{cases} -2(\theta^v_{ij}+\pi)/\pi \ if \quad \theta^v_{ij} \ < -\pi/2 \\ -1 \qquad\qquad\quad if\ -\pi/2 < \ \theta^v_{ij} \ < \ 0 \\ +1 \qquad\qquad\quad if \quad 0 \ < \ \theta^v_{ij} \ \le \pi/2 \\ -2(\theta^v_{ij}-\pi)/\pi \ if \quad \theta^v_{ij} \ > \ \pi/2 \end{cases} .$$

The output are the velocity in the next time step in polar coordinates as show Fig. 2(b).

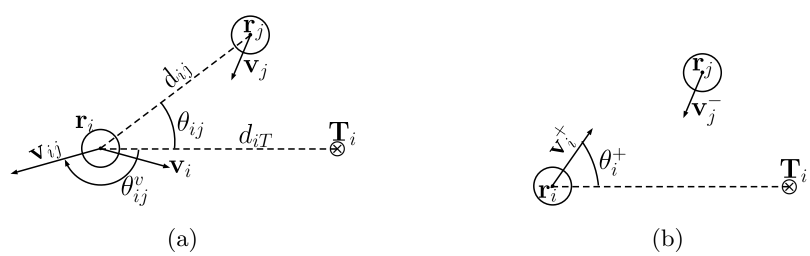


Fig. 2: (a) Basic quantities needed for defining the input vector ($\boldsymbol{\xi}$). (b) Polar coordinates of the future velocity.

The nonparametric neural network We call $\mathbb{E} = \{\boldsymbol{\xi}(t), \boldsymbol{\zeta}(t)\}$ the experimental set of state/action examples having data points for each time step t.

Each one of the two components of the output vector $\boldsymbol{\zeta}(t)$ (eq. 2) will we approximated by one neural network with output ${}^{\mu}O\ :\ \mathbb{R}^6 \rightarrow \mathbb{R}$, where $\mu = 1, 2$ indicates its polar components, i.e., the speed (v^+_i) and the angle (θ^+_i) respectively.

The neural network we choose is the generalized regression neural network (GRNN) [4], which is a type of radial basis function network [5].

The GRNN is a universal interpolator based on nonparametric regression. The basic idea is that when trying to predict the output for a new input, the data examples are used in the following way: first, the distance between the new input and the data inputs is calculated, then the corresponding data outputs are weighed with a kernel function, depending on that distance, and averaged. In other words, the data outputs of closer data inputs are used for interpolating the new output.

In what follows we explain this concept explicitly for a network with one dimensional output (O). Suppose a training family of ordered pairs $\{\boldsymbol{\xi}_n, \zeta_n\}_{n \leq N}$, then:

$$O(\boldsymbol{\xi}) = \frac{\sum_{n=1}^{N} \zeta_n K(\boldsymbol{\xi}, \boldsymbol{\xi}_n)}{\sum_{n=1}^{N} K(\boldsymbol{\xi}, \boldsymbol{\xi}_n)} \tag{3}$$

where

- $O(\boldsymbol{\xi})$ is the prediction value of an arbitrary input vector $\boldsymbol{\xi}$.
- ζ_n is the output of example n corresponding to the input vector $\boldsymbol{\xi}_n$.
- $K(\boldsymbol{\xi}, \boldsymbol{\xi}_n) = e^{-l_n/2\sigma^2}$ is the radial basis function kernel that weighs the contribution of the nth output example in order to predict the new output.

Where $l_n = (\boldsymbol{\xi} - \boldsymbol{\xi}_n)^T(\boldsymbol{\xi} - \boldsymbol{\xi}_n)$ is the square distance between data examples $\boldsymbol{\xi}_n$ and the input vector $\boldsymbol{\xi}$.

Once we have a proper set of N patterns, the only degree of freedom in this neural network is the so-called spread (σ), which can be taken as a scalar value for all examples and variables of the input vector.

3 Simulations

In this section we describe how the spread (σ) of both GRNN's was calibrated and we present results showing that with the proposed approach we can simulate a configuration of a pedestrian avoiding an obstacle.

Simulation scheme At each time step of the simulation, the input state of the simulated particle is calculated and both GRNN's will provide the speed (v^+) and angle (θ^+) as the outputs corresponding to the input state, then using $[v^+cos(\theta^+), v^+sin(\theta^+)]$ as the desired velocity $\mathbf{v}_i^s$ we can update the position $\mathbf{r}(t)$ of the particle at time t as follow:

$$\mathbf{r}^s(t + \Delta t) = \mathbf{r}^s(t) + \mathbf{v}^s(t)\Delta t,$$

and this process continues until the simulated particle reach the final target.

As stated in the previous section, there is only one free parameter for each GRNN: the spread (σ). In the next section we specify how this parameter was determined.

Calibrating the GRNN For the determination of the spread, we consider the 13 experimental trajectories $\{\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_{13}\}$ and proceed with a leave-one-out cross-validation. We consider the spread of both GRNN equal and define a error functions between simulated and experimental trajectories based on the minimum distance to the obstacle, $E_d(\sigma)$.

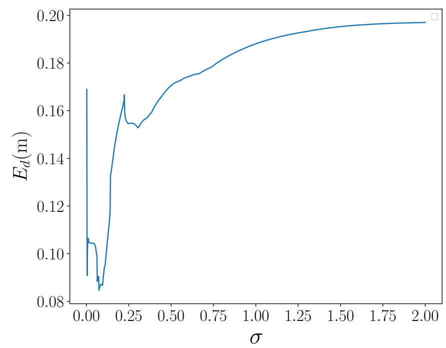


Fig. 3: Measures of the error for the simulated trajectories comparing with the experimental ones as a function of the GRNN parameter σ.

Figure 3 shows the results. The optimum spreads found were: $\sigma = 0.074$ with an error $E_d = 0.08$ m.

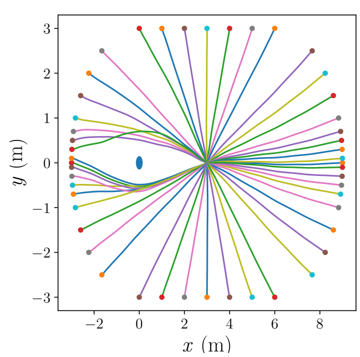
Results Using the σ obtained in the calibration and the set of 26 trajectories as data we analyze the performance simulating several particles starting around the goal, some of them having to avoid the obstacle.

The system to be simulated consists of a fixed obstacle located and a final target for all particles. Forty-eight new particles were simulated once at a time, with initial positions at 6 m from target as shown in Fig. 4 (a).

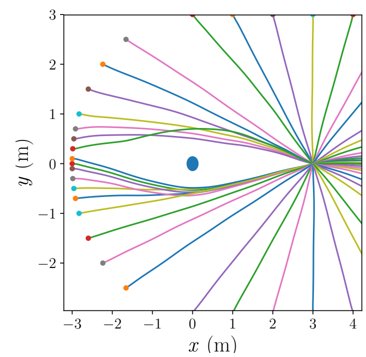
In Fig. 4 (b) the smoothness and continuity of the trajectories with respect to the initial positions can be seen except for one trajectory that slightly crosses over other neighbors' trajectories. This crossing is also observed in the experiments as is shown in Fig. 1. It should be noted that only potentially colliding trajectories produce a detour for avoiding the obstacle, while the rest of the particles describe straight trajectories toward the target.

4 Conclusions and perspective

We propose a data-driven model for a simple configuration of one pedestrian and one fixed obstacle using generalized regression neural network (GRNN). The principal advantage of this neural network is that it only has one free parameter that was calibrated using a leave-one-out cross-validation.



(a)



(b)

Fig. 4: Simulated trajectories with $\sigma = 0.074$. (a) Complete view of the simulated system. (b) Zoom over the avoidance region.

In addition, the method presented allows us to reproduce several configurations with one pedestrian walking freely or avoiding a narrow medium distance obstacle. The method is invariant under rotations.

The presented data-driven method could be extended to simulate more complex configuration considering two or more pedestrians.

Acknowledgements

The authors acknowledge financial support via project PID2015-003 (Agencia Nacional de Promoción Científica y Tecnológica, Argentina; Instituto Tecnológico de Buenos Aires; Urbix Technologies S.A.) and from ITBACyT-2018-42 (Instituto Tecnológico de Buenos Aires).

References

1. Martin, R.F., Parisi, D.R.: Pedestrian collision avoidance with a local dynamic goal. In: Proceedings of the PED2018 Conference (in progress) (2018)
2. Moussaïd, M., Helbing, D., Theraulaz, G.: How simple rules determine pedestrian behavior and crowd disasters. Proceedings of the National Academy of Sciences **108**(17), 6884–6888 (2011)
3. Seitz, M.J., Bode, N.W., Köster, G.: How cognitive heuristics can explain social interactions in spatial movement. Journal of The Royal Society Interface **13**(121), 20160,439 (2016)
4. Specht, D.F.: A general regression neural network. IEEE transactions on neural networks **2**(6), 568–576 (1991)
5. Park, J., Sandberg, I.W.: Universal approximation using radial-basis-function networks. Neural computation **3**(2), 246–257 (1991)