A neural network for the prediction of wave reflection from coastal and harbor structures

Barbara Zanuttigh a,⁎, Sara Mizar Formentin a, Riccardo Briganti b

a University of Bologna, DICAM, Viale Risorgimento 2, 40136 Bologna, Italy
b Faculty of Engineering, University of Nottingham, UK

A R T I C L E   I N F O

Article history:
Received 29 July 2012
Received in revised form 13 May 2013
Accepted 21 May 2013
Available online 15 June 2013

Keywords:
Wave reflection
Artificial neural network
Database
Coastal structures
Empirical formulae
Sensitivity analysis

A B S T R A C T

This paper presents the development of an Artificial Neural Network for the prediction of the wave reflection coefficient from a wide range of coastal and harbor structures. The Artificial Neural Network is trained and validated against an extensive database of about 6000 data, including smooth, rock and armor unit slopes, berm breakwaters, vertical walls, low crested structures, oblique wave attacks. The structure and data included in this database, as well as the approach used in this paper, follow the work done on wave overtopping within the CLASH project.

In this new Artificial Neural Network 13 input elements are used to represent the physics of the reflection process taking into account the structure geometry (height, submergence, straight or non-straight slope, with or without berm or toe), the structure type (smooth or covered by an armor layer, with permeable or impermeable core) and the wave attack (water depth, wave height, wave length, wave obliquity, directional spreading). The selection of the input elements and of the algorithms used in the network is described based on an in-depth sensitivity analysis of the network performance.

The accuracy of the network is quite satisfactory, being the average root mean squared error lower than 0.04. This value is consistent between the Artificial Neural Network calibrated on the original dataset and the one calibrated on boot-strapped datasets in which data reliability and structure complexity are considered.

The performance of the network is compared for limited datasets with selected available literature formulae proving that this approach is able to estimate the experimental reflection coefficients with greater accuracy than the empirical formulae calibrated on these same datasets.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

Wave reflection from coastal and harbor structures may compromise structure stability, due to the induced intense scour at the structure toe, and may endanger harbor access, due to sea states at the entrance. Despite the relevance of accurately estimating the wave reflection coefficient \( K_r \) for design purposes, the research focused on wave reflection only is fairly limited and it is very often a by-product of research on structure stability, wave overtopping or wave transmission. Several empirical formulae for predicting \( K_r \) exist indeed, but most of them were developed on limited datasets related to specific structures, i.e. smooth and impermeable structures (Battjes, 1974; Miche, 1951; Seelig and Ahrens, 1981; Ursell et al., 1960), permeable structures (Losada and Giménez-Curto, 1981; Numata, 1976; Seelig and Ahrens, 1981), permeable structures with an impermeable core (Postma, 1989), armor units such as tetrapods, accropods or other artificial elements (Allsop and Hettiarachchi, 1989; Murtay et al., 2006). Specific studies were carried out also on prototypes (Davidson et al., 1996) and special shaped structures, such as Low Crested Structures, hereafter LCSs (Calabrese et al., 2008), and vertical walls (Takahashi, 1996). Recently, based on the data collected and the format agreed within the F5S EC-funded CLASH project (Van der Meer et al., 2008), Zanuttigh and Van der Meer (2008) gathered a database which collects various types of structures (emerged and submerged; with and without berms; permeable and impermeable; etc.). They developed a formula for the prediction of \( K_r \), which can be extended to structures with berms or subjected to oblique waves (Zanuttigh and Lykke Andersen, 2010; Zanuttigh et al., 2009).

Besides of these empirical formulae, wave reflection can be estimated using numerical models. Many studies of wave reflection at coastal structures have been carried out in relation to harbor resonance with both phase-averaged (e.g. Elchahal et al., 2008; Isaacson and Qu, 1990) and phase resolving models (Shi et al., 2003). Depth integrated models based on both Non-Linear Shallow Water Equations and Boussinesq-type equations have been used to assess wave transformations induced by coastal structures (e.g. Johnson et al., 2005; Wurjanto...
and Kobayashi, 1993). For a near-field analysis, depth resolving models are preferred. Recently, solvers based on the Reynolds Averaged Navier Stokes and a Volume of Fluids technique for the free surface tracking (RANS-VOF) models have been used (Lara et al., 2006; Losada et al., 2008). Notwithstanding faster computing resources, they are still time-consuming and may also lead to overestimation of wave reflection (Zanuttigh et al., 2009).

In conclusion, at present a sufficiently accurate and yet efficient prediction model, valid for a wide range of structures, is not available to be implemented in a design support system.

In alternative to traditional techniques Artificial Neural Networks (ANNs) offer flexibility and accuracy. ANNs have been successfully used in coastal engineering in a wide range of applications. Among them it is worth mentioning the prediction of time series of wave parameters in a specific place of interest (Browne et al., 2006; Deo et al., 2001; Kalra et al., 2005; Makarynskyy et al., 2005a,b; Rao and Mandal, 2005; Tsai et al., 2002 among others), data reconstruction (Makarynskyy, 2005), the analysis of interdependencies between wave parameters (Agarwal and Deo, 2004; Deo et al., 2002) and the improvement of the accuracy of numerical models results (Makarynskyy, 2005).

In the specific case of wave-structure interaction, wave overtopping is by far the subject in which ANNs have been applied more often in recent years. Early studies (Medina, 1999; Medina et al., 2002, 2003) were followed by the results of the CLASH project: the ANNs developed by Van Gent et al. (2007) and by Verhaeghe et al. (2008) are currently standard methods for overtopping computation.

Besides wave overtopping, ANNs were set-up also to predict structure stability (Mase et al., 1995), forces on vertical structures (Van Gent and Van den Boogaard, 1998), wave transmission at low crested structures (Panizzo and Briganti, 2007). A very recent application of ANN to wave reflection from a specific type of caisson breakwater has also been carried out (Garrido and Medina, 2012).

This paper aims to propose an ANN for the estimation of the wave reflection coefficient from coastal and harbor structures under a variety of wave conditions, by means of an extended version of the experimental database prepared by Zanuttigh and van der Meer (2008).

In Section 2, the database used to train and test the ANN is presented. Section 3 describes the ANN architecture and the selected input elements; the ANN results and performance are illustrated in Section 4. Sections 5 and 6 present an in-depth sensitivity analysis for a limited set of data. In Section 8 a bootstrapping resampling based on appropriate weight factors attributed to each test is proposed, in analogy to the CLASH approach. Conclusions are finally drawn in Section 9, including a short discussion on the overall performance and limitations.

2. Description of the data

The ANN is expected to satisfy two requirements: by one hand, it has to be as much versatile as possible in order to be able to work with several different types of structures; by the other hand, its prediction must be reliable.

The database employed to train, test and validate the new wave reflection ANN is almost (97%) the same as the wave reflection database described in Zanuttigh and Van der Meer (2008). This database, in turn, includes part of the DELOS wave transmission database (Van der Meer et al., 2005), part of the CLASH wave overtopping database (general presentation in Steendam et al., 2004; armor unit data in Bruce et al., 2006), data acquired from model testing in European facilities (among the others, Lissev, 1993), field measurements (at Elmer, UK, see Davidson et al., 1996), recent tests on low-crested structures (Cappietti et al., 2006) for a total of around 4000 data. This database was supposed to be wide and diverse enough to let the ANN achieving the first requirement of versatility. However, it lacks of data characterized by high values of the wave reflection coefficient ($K_r$). Therefore, new data of seawalls, derived from Requejo et al. (2002) and kindly provided by Kortenhaus (private communication, 2012), were included.

The ANN second requirement of reliability needs a consistent and accurate database. A large part of the database used here corresponds to the original wave reflection database by Zanuttigh and Van der Meer (2008). This, in turn, derives from the wave overtopping database of the CLASH project, which has been obtained after a careful studying and screening of each dataset before gathering together the data. This same process of screening each dataset and each test within the dataset was carried out by Zanuttigh and Van der Meer (2008) and has been performed here with the additional dataset, in order to maintain the database quality.

The updated reflection database set-up for the ANN includes a total of 5781 data, whose synthesis is given in Table 1.

The format of the database follows precisely the structure of the wave overtopping database gathered within the CLASH project (van der Meer et al., 2008). In CLASH, for each test, three groups of parameters were defined: 3 general parameters, 11 hydraulic parameters and 17 structural parameters. The general parameters consist of an unique name for each test and the reliability and complexity factors, RF and CF. The hydraulic parameters describe the wave characteristics and the measured overtopping, whereas the structural parameters describe the tested structure (as depicted in Fig. 1).

In Table 2 the type and the number of parameters included in the original CLASH database and in the wave reflection database (following Zanuttigh and Van der Meer, 2008) are shown. Table 2 also specifies the parameters involved in the new wave reflection ANN here presented and in the existing wave overtopping (Verhaeghe et al., 2008) and wave transmission ANNs (Panizzo and Briganti, 2007).

The parameters employed for the definition of the input vector of the ANN (see Fig. 1 and Table 2) are described below:

- $h$: water depth in front of the structure, [m];
- $H_{m0, r}T_m$ : wave main height [m] and spectral period [s], at the toe;
- $h_t$: toe submergence, [m]; in absence of toe, it is equal to $h$, the water depth in front of the structure;
- $h_b$: berm submergence, [m]; if the structure has neither berms nor toes, it is equal to $h$;
- $\alpha_c$: slope of the lower part of the structure, below the berm if it is present, [°];
- $\alpha_{nc}: \text{mean slope within the run-up and run-down zone}$, including the berm; the mean angle $\alpha_{nc}$ is defined between $\pm 1.5\text{SH}_{n, 0.1}$; in case of straight slopes, $\cot \alpha_{nc} = \cot \alpha_c$;
- $\gamma_{c}$: armor roughness coefficient [−];
- $D_{n, 50}$: nominal armor stone diameter, [m]; in case of armor units, it is the representative mean dimension; for smooth and impermeable structures, its value is ideally set to 0;
- $R_w$: breakwater freeboard, eventually comprehensive of the crown sea wall, [m];
- $A_c$: breakwater freeboard non-comprehensive of the eventual crown sea wall, [m];
- $b$: berm width, [m];
- $G_c$: breakwater crest width, [m];
- $m$: foreshore slope, defined as the cotangent of the angle of the slope itself, [°];
- $\beta$: angle of deviation from the perpendicular wave attack direction, [°];
- $\text{spreading}$: wave directional spreading, [°];
- $RF$, $CF$: “reliability factor” and “complexity factor” respectively: the former associates a degree of reliability to each test; the latter describes the structure geometry and its relative complexity;
- $K_r$: wave reflection coefficient, [−].
It is important to note that, for almost all datasets, the computed values of $K_r$ and not the measured data at the wave gauges were available, so that it was impossible to perform directly a homogeneous spectral wave analysis over the whole database. The methods to derive $K_r$ differ from dataset to dataset, the most widely adopted being by Mansard and Funke (1987), Zelt and Skjelbreia (1992) and Hughes (1993).

### 3. The Artificial Neural Network

The final form of the ANN for the prediction of $K_r$ is presented here. Its architecture consists of 13 input elements and 40 hidden neurons; the output neuron is naturally $K_r$. This section is organized as follows: Section 3.1 provides the network architecture; in Section 3.2, the 13 input elements (corresponding to 13 non-dimensional parameters) are listed and motivated; the ranges of values of the parameters are summarized in Table 3.

#### 3.1. Artificial Neural Network architecture

The architecture of the model has been optimized through a sensitivity analysis to the input elements (derived from the physical interpretation of the reflection process, see Section 5) and to the neural model specific functions (for example, the choice of the best and

---

**Table 1**

Tests included in ANN training, testing and validating database. Where reference is not given, data have been kindly provided by private communications.

<table>
<thead>
<tr>
<th>Database section</th>
<th>Structure type</th>
<th>Samples</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LCs</td>
<td>830</td>
<td>Seabrook and Hall (1998), Ruol et al. (2004), Van der Meer et al. (2005), Zanuttigh and Lamberti (2006), Cappietti et al. (2006)</td>
</tr>
<tr>
<td>B</td>
<td>Rock impermeable straight slopes</td>
<td>Breakwaters</td>
<td>198</td>
</tr>
<tr>
<td>C</td>
<td>Armor units straight slopes</td>
<td>Accropods</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Antifer</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tetrapods</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Core-Locs</td>
<td>145</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Cubes</td>
<td>234</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Haros</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Dolos</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Wave-walker blocks</td>
<td>63</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Acquareefs</td>
<td>1063</td>
</tr>
<tr>
<td>D</td>
<td>Smooth straight slopes</td>
<td>Breakwaters</td>
<td>189</td>
</tr>
<tr>
<td></td>
<td></td>
<td>T-blocks</td>
<td>23</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Vilvord and Haringman blocks</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Mobile flood defence system</td>
<td>170</td>
</tr>
<tr>
<td>E</td>
<td>Structure with combined slopes and berms</td>
<td>Rocks</td>
<td>99</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rocks and asphalt</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tetrapods</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Antifer</td>
<td>34</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Smooth</td>
<td>265</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Block revetments</td>
<td>57</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Vilvord rocks and basalt</td>
<td>8</td>
</tr>
<tr>
<td>F</td>
<td>Seawalls</td>
<td>Vertical wall</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Vertical wall with toe</td>
<td>57</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Vertical wall with foreshore</td>
<td>27</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Caisson and circular caisson</td>
<td>42</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Vertical porous breakwater</td>
<td>59</td>
</tr>
<tr>
<td>G</td>
<td>Oblique attacks</td>
<td>Rocks LCs</td>
<td>84</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Large crested rocks</td>
<td>495</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Short crested rocks</td>
<td>233</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Cubes</td>
<td>569</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Smooth LCS</td>
<td>84</td>
</tr>
</tbody>
</table>

**Fig. 1.** Structural and hydraulic parameters involved in the wave reflection and CLASH databases.
The ranges of values of the ANN 13 input elements.

Table 3
The ranges of values of the ANN 13 input elements.

| Parameter | Complete Database #5781 | Min | Max | Min | Max | Min | Max | Min | Max | Min | Max | Min | Max | Min | Max | Min | Max |
|-----------|-------------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Hm,0,deep | [m]                     | 0.001 | 0.121 | 0.001 | 0.094 | 0.006 | 0.073 | 0.004 | 0.121 | 0.006 | 0.071 | 0.009 | 0.079 | 0.004 | 0.116 | 0.024 | 0.082 |
| Bt        | [m]                     | 0.008 | 1.892 | 0.017 | 0.719 | 0.063 | 0.340 | 0.020 | 0.829 | 0.008 | 1.892 | 0.038 | 0.330 | 0.014 | 0.438 | 0.083 | 0.445 |
| Hm,0,t    | [m]                     | -8.087 | 25.391 | -3.922 | 11.714 | 0.309 | 7.543 | -8.087 | 8.449 | 0.000 | 25.391 | 0.262 | 9.589 | 0.083 | 12.051 | -0.658 | 1.902 |
| cot ϑd    |                         | 0.000 | 7.000 | 0.820 | 5.000 | 2.000 | 6.000 | 1.250 | 2.000 | 1.000 | 4.000 | 1.500 | 7.000 | 0.000 | 1.732 | 2.000 | 3.000 |
| tanb      |                         | 0.340 | 1.000 | 0.350 | 1.000 | 0.550 | 0.600 | 0.380 | 0.500 | 0.800 | 1.000 | 0.380 | 1.000 | 0.340 | 1.000 | 0.400 | 1.000 |
| tanθ       |                         | 0.000 | 6.537 | 0.178 | 3.956 | 0.140 | 6.537 | 0.243 | 2.281 | 0.000 | 0.000 | 0.116 | 4.942 | 0.000 | 1.323 | 0.000 | 0.834 |
| ∆L         |                         | 0.000 | 2.238 | 0.000 | 2.484 | 0.000 | 0.085 | 0.000 | 2.238 | 0.000 | 0.000 | 0.000 | 0.467 | 0.000 | 0.885 | 0.029 | 0.170 |
| β          |                         | 0.000 | 83.490 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 83.490 |
| Pow        | [m3/sm]                 | 0.000 | 1.024 | 0.000 | 1.322 | 0.000 | 0.000 | 0.000 | 1.222 | 0.000 | 0.000 | 0.000 | 1.024 | 0.000 | 0.000 | 0.000 | 0.000 |
| cotuθcrl   |                         | -1.972 | 6.091 | -0.451 | 0.972 | 0.000 | 0.000 | 0.000 | 6.091 | 0.000 | 0.000 | 0.000 | -1.972 | 2.291 | -0.811 | 0.904 | 0.000 |
| spreading  |                         | 0.000 | 10.012 | 0.570 | 5.531 | 2.000 | 6.000 | 1.250 | 2.000 | 1.000 | 4.000 | 0.894 | 10.012 | 0.004 | 4.840 | 2.000 | 3.000 |
| m          |                         | 0.000 | 50000 | 0.000 | 1.000 | 0.000 | 0.000 | 0.000 | 1.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 50000 |

most suitable training algorithm, the number of hidden neurons definition, etc., see Section 6). It is built in the MATLAB environment with the following fundamental characteristics:

- multilayer network, based on a "feed-forward back-propagation" learning algorithm;
- static network (absence of delays and feedbacks);
- structure and layers: the input vector with 13 elements (see also Section 3.2), 1 hidden layer with 40 hidden neurons and 1 output neuron, the wave reflection coefficient \( K_r \);
- hidden neurons transfer function: hyperbolic tangent sigmoid transfer function;
- output neuron transfer function: linear transfer function;
- data structures: the arrays of data can be provided to the network.
as "concurrent vectors", i.e. rows of a matrix in a random order; as long as a neural network is static, the way the arrays are provided to the ANN (i.e. subsequent or concurrent way) has no relevance for the simulation while for the training phase it is very important since the connections weights are updated accordingly; in fact, connections weights may be updated either at the end of each training epoch or within each epoch for each new test:

- training style: “batch training”, connections weights and biases are updated at the end of each training epoch, just once the ANN has read all the input data;
- training algorithm: Levenberg–Marquardt algorithm;
- learning algorithm: momentum gradient descent back-propagation algorithm.

In Fig. 2 the ANN logic layout is schematized, while Fig. 3 shows the ANN architecture and characteristics using MATLAB notation style.

The notation "IW" means “Input Weights”, and refers to the weights attributed to the connections among the inputs and the hidden neurons during the training process, while "LW" means “Layer Weights”, i.e. the weights associated to the connections among the output neuron and the hidden neurons. The symbol “b” relates to the “Biases”, two elements which belong, respectively, to the hidden layer ([b(1)]) and to the output layer ([b(2)]). These are defined by the ANN itself during the training as well. The symbols in the grey rectangles represent the transfer functions, respectively the sigmoid and the linear one; the plus simply indicate the algebraic sum between weights and bias.

3.2. Input elements

The parameters involved in the input vector should include a synthesis of sea and structural characteristics. The choice of the parameters was initially based on the following relations (Eqs. (1) and (2)):

- Zanuttigh and Van der Meer (2008):

$$K_r = \tanh \left( a \xi_0 \right),$$

where

$$a = 0.167 \left[ 1 - \exp \left( -3.2 \gamma_f \right) \right]$$

and

$$\xi_0 = \frac{\tan \gamma_f}{\sqrt{H_{m0} / \lambda_{\text{ref}}}}$$

- Muttray et al. (2006):

$$K_r = \frac{1}{1.3 + 3h \left( \frac{2}{2h - 1} \right)}$$

Besides the parameters included in Eqs. (1) and (2), additional parameters have been selected to represent more complex structure geometries, i.e. including a toe or a berm or a crown wall, special structures, such as vertical slopes and circular caissons, and also oblique wave conditions. Each parameter is added to the first set of parameters derived from Eqs. (1) and (2) and a sensitivity test of the ANN performance is carried out considering the extended input dataset. Altogether, the following non-dimensional 13 parameters have been finally chosen as elements of the input vector (values summarized in Table 3):

I. $H_r / 2 m_0 / m_0$: it is proportional to the wave steepness and it is part of the breaking parameter, which represents the breaker level of energy;

II. $h_0 / L_m - 1.0$: it accounts for shoaling effects associated to incident waves;

III. $\cot \alpha$: off-shore structure slope in the run-up area; together with $I$, it completes the description of $\xi_0$. The cotangent form has been preferred to the tangent to prevent infinite values of the tangent in presence of seawalls (see Tables 1 and 2, database “F”), i.e. when $\alpha = 90^\circ$;

IV. $\gamma_f$: armor layer roughness factor. It is an index of wave energy dissipation during the run-up process. The higher the values of $\gamma_f$ the smoother and the more reflective the structure.

V. $R_c / H_m$: the lower the relative crest freeboard, the greater the wave transmission and the lower the $K_r$. Though it apparently doesn’t appear in Eqs. (1) and (2), it plays a key role in the formulation of the ranges of validity associated to these formulae.

VI. $D_{50} / H_m$: this term essentially represents the wave pressure inside the structure pores; it is involved in the definition both of design conditions and of the ranges of validity for existing $K_r$ formulae (Calabrese et al., 2008; Davidson et al., 1996; Zanuttigh and Van der Meer, 2008). In addition, Panizzo and Briganti (2007) used this parameter as input element for ANN they developed to predict the values of the transmission coefficient for LCSs.

VII. $\beta$: this parameter is essential to describe the effects of oblique wave attacks. The greater the wave obliquity the greater the structure surface exposed to wave action and therefore the greater the wave dissipation and the lower the wave reflection. Usually, the reductions of wave run-up and wave overtopping due to wave obliquity are represented by the factor $\gamma_f$ (Eurotop, 2007), that depends on $\beta$ through coefficients still needing calibration. The significance of $\gamma_f$ for predicting $K_r$ has been already shown by Zanuttigh and Lykke Andersen (2010). In order to avoid the use of coefficients that may be inaccurate, it has been selected to use directly $\beta$ rather than $\gamma_f$.

Fig. 2. Logic layout of the wave reflection ANN.
VIII. \( G_c/L_m - 1.0 \): this parameter is introduced to represent any possible reflected wave phase delay from the upper part of the breakwater.

IX. \( B/L_m - 1.0 \): this parameter, as well X and XI, is used to describe the effects induced by a berm. It is conceptually similar to VII, in order to account for the phase delay of waves reflected from the structure down-slope and from the berm.

X. \( h_o/H_m,0 \): this parameter is essentially a modified breaking index to account for the process of waves breaking on the berm, dissipating energy and reducing the run-up process over the structure upper slope.

XI. \( \cot \alpha_{incl} \): also the knowledge of the angle \( \alpha_{incl} \) is important for the network as long as it differs from \( \alpha_b \), i.e. when the structure has a berm, because of the difference in estimating \( \xi_b \).

XII. \( m \): a foreshore in front of a breakwater might induce additional reflection while waves travel from offshore to the structure toe.

XIII. spreading: the directional wave spreading tends to increase the effects induced by wave obliquity, and therefore the greater the directional spreading the lower the wave reflection.

3.3. Evaluation of the network performance

The training phase consists of three components: the proper training, the testing and the validation. The full database is randomly divided into three parts: the training database (60% of the overall data) is actually employed to “teach” the model the input-output relations, while the testing and validating database (the remaining data equally divided for the two purposes) are useful to compute the model errors during the same training phase. The training phase stops when the mean squared error (\( \text{rmse} \)) increases instead of monotonously decreasing (i.e. the MATLAB function “early stopping” is employed).

Since the selection of the data for training, testing and validation is random, the combined training and subsequent ANN simulation is a stochastic process. The results of each combined training and simulating run of the ANN are statistically independent events, since the ANN is re-initialized each time. Therefore, in order to describe properly the ANN performance, these combined runs were repeated many times and the root mean squared error (“\( \text{rmse} \)” derived from each run was stored and finally statistically evaluated as follows:

\[
\text{rmse} = \frac{1}{40} \sum_{i=1}^{40} \left( \frac{1}{5781} \sum_{j=1}^{5781} \left( K_{i,j} - K_{i,\text{ANN}} \right)^2 \right) = \frac{1}{40} \sum_{i=1}^{40} \text{rmse}_i \quad (3)
\]

In Eq. (3), index \( i \) denotes the \( i \)-th training–simulation process, index \( j \) refers to the \( j \)-th data, 40 is the number of training–simulations and 5781 is the total number of data.

In order to identify the optimal number of combined training and simulation processes, an analysis of the network performance variability was carried out. Diagrams of Fig. 4 show, respectively, the \( \text{rmse} \) value and its standard deviation with increasing number of training–simulations. It can be observed that the average value of \( \text{rmse} \) stabilizes around an almost constant value of 0.038 for a number of training–simulation processes greater than 30, while its standard deviation continues to monotonically decrease till reaching the value of 0.004 for a number of training–simulation processes greater than 80. Considering that between 40 and 80 simulations the value of the standard deviation decreases just from 0.005 to 0.004, it has been decided to choose 40 as the optimal number to characterize the network performances. Therefore, the results illustrated in the following sub-sections refer to the mean values obtained from 40 training–simulation processes.

4. Model results

This section describes the most relevant results obtained from the ANN presented in the previous Section 3, once trained on the complete database (see Section 2).

In Section 4.1 the outcomes of the ANN (i.e. the weights attributed to the connections between the neurons and the bias) are described,
while the performance of the ANN is discussed in Section 4.2. It is qualitatively shown by the comparison of the predicted values of \( K_{r,\text{ANN}} \) with the measured ones \( K_{r,s} \) (Fig. 5) and through a frequency distribution histogram of the same quantity \( e \) (Fig. 6).

Quantitative estimate of the ANN accuracy is provided through the root mean squared error (rmse, Eq. (3)) and the Willmott index (WI, see Willmott, 1981). Error analysis is carried out in Section 4.3.

4.1. Weights and biases

The fundamental products of the trained ANN are the weights and the bias elements, which allow to use the network as a predictive model. These elements can be matrices, vectors or scalars and contain the values attributed to the connections among neurons by the model itself during the training process.

In this case, we have 4 elements: the matrix of “Input Weights” (IW, of dimensions 40 \( \times \) 15) to connect hidden neurons to inputs, the matrix of “Layer Weights” (LW, a single row of dimensions 1 \( \times \) 40) for output neuron—hidden neurons connections, the bias vector (40 \( \times \) 1) for hidden layer and the bias scalar (1 \( \times \) 1) for the output layer. 681 “parameters” (i.e. weights associated to connections) are employed overall. It is worthy to note that the 40 \( \times \) 15 matrix dimensions relate to the number of hidden neurons connections, the bias vector (40 \( \times \) 1) for hidden layer and the bias scalar (1 \( \times \) 1) for the output layer. 681 “parameters” (i.e. weights associated to connections) are employed overall. It is worthy to note that the 40 \( \times \) 15 matrix dimensions relate to the number of hidden neurons connections, the bias vector (40 \( \times \) 1) for hidden layer and the bias scalar (1 \( \times \) 1) for the output layer.

The weight matrices are provided in Appendix A (Section 12). These have been randomly chosen after the 500 ANNs provided by 500 corresponding bootstrapping resamples of the database developed in order to assess the ANN uncertainty (see Section 8).

4.2. Results

In Fig. 5, the computed values of \( K_{r,s} \), i.e. \( K_{r,\text{ANN}} \), are compared with the measured values, \( K_{r,s} \). The central line represents the bisector, i.e. the perfect correspondence among predicted and experimental values, while the external lines represent the 95% confidence boundaries. The graph shows the pretty good agreement of computations and measurements; the confidence interval is quite narrow and, moreover, the ANN appears to provide the results with a good degree of symmetry, as it can be appreciated also by the histogram in Fig. 6.

The quantitative estimate of the ANN accuracy is provided through the above-mentioned rmse, the Willmott index (“WI”, Willmott, 1981), which takes into account also the error distribution symmetry and the coefficient of determination \( R^2 \), which is defined as follows:

\[
R^2 = 1 - \frac{\sum_{j=1}^{5781} (K_{r,\text{ANN}j} - K_{r,sj})^2}{\sum_{j=1}^{5781} (K_{r,sj} - K_{r,sj})^2}, \quad \text{where } K_{r,sj} = \frac{1}{5781} \sum_{j=1}^{5781} K_{r,sj} \tag{4}
\]

The mean and standard deviation obtained from 40 training-simulations are reported in Table 4. The rmse value of 0.038 is particularly good, especially if compared to existing formulæ (see Section 7), characterized by larger values of rmse and associated to well defined structures typologies and therefore more restricted datasets (Zanuttigh and Lykke Andersen, 2010; Zanuttigh and Van der Meer, 2008). The very high WI value of 0.985 denotes that the error distribution is not just moderate, but also satisfactory symmetric. The values of standard deviation of rmse, WI and \( R^2 \) are all around \( 10^{-3} \), denoting a good ANN stability, in spite of the random data-selection processes (training and testing). The model is therefore consistently accurate and reliable.

4.3. Error analysis

To assess the ANN performance in greater details, the \( K_{r,\text{ANN}} \) values affected by larger errors are analyzed here. For this purpose the “threshold” value \( e = |0.15| \) has been set since it represents, on an

![Fig. 5. Comparison among \( K_r \), predicted values (\( K_{r,\text{ANN}} \), ordinate) and corresponding \( K_r \), experimental values (\( K_{r,s} \), abscissa).](image)

![Fig. 6. Difference \( e = K_{r,s} - K_{r,\text{ANN}} \) frequency distribution histogram.](image)

<table>
<thead>
<tr>
<th></th>
<th>Mean</th>
<th>Stand. dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>rmse</td>
<td>0.038</td>
<td>0.003</td>
</tr>
<tr>
<td>WI</td>
<td>0.985</td>
<td>0.003</td>
</tr>
<tr>
<td>( R^2 )</td>
<td>0.943</td>
<td>0.006</td>
</tr>
</tbody>
</table>

Table 4
40 simulations rmse, WI and \( R^2 \) average values and corresponding standard deviation; final model.
average, a percentage relative error of approximately 50%, being the average value of \( K_{rs} \) ≈ 0.26 (see Table 5). Errors lower than 0.15% are supposed to fall within the random uncertainty, while greater errors deserve a more accurate analysis.

Subsequently, the errors obtained from the 40 simulations of the ANN have been analyzed. Fig. 7 shows all the data the ANN associated to \( e \geq 0.15 \) at least once during these 40 simulations. The data repeatedly affected by “large” errors are immediately visible, thanks to the concentration of error points aligned on its corresponding indices (to the left) or values of \( K_{rs} \) (to the right).

If the same test is systematically affected by a “large” error, it may imply either that the test itself is less reliable (for example, due to measurement errors especially in case of low \( K_{rs} \)) or inaccurate methodology (analysis) or that the ANN does not correctly learn the input-output relation for the test. Based on the diagram to the left of Fig. 7, it is clear that “large” errors occur especially for specific values of the indexes, which correspond to structures with berms or seawalls (indexes values >4500) and smooth impermeable structures (3000 ÷ 3500).

Differently, based on the plot to the right of Fig. 7, there is no evident specific concentration of “large” errors depending on the values of \( K_{rs} \). The ANN however tends to overestimate the very low values of \( K_{rs} \) \( (K_{rs} < 0.1) \) and to underestimate the very high values of \( K_{rs} \) \( (K_{rs} > 0.85) \), showing the typical ANN problems occurring with extrapolation.

From a quantitative viewpoint, the ANN has made 508 “large” errors over the 40 simulations (i.e., altogether over more than 230,000 data), i.e. a frequency of 0.2% of “large” errors \((e_{large})\) is estimated. In order to quantify the average “large” error provided by the ANN and compare it to the global ANN performance, two relative error indicators are defined in Eqs. (5) and (6). The results are synthesized in Table 5, showing that the ANN would produce errors larger than 80% \( (\bar{e}_{large} = 83\%) \) just “twice over 1000 data” (0.2%), while on average the ANN produces errors approximately lower than 10% \( (\bar{e}_{s} = 8.3\%) \).

\[
\bar{e}_{s} = \frac{\bar{e} \cdot 100}{K_{rs}} , \quad \text{where} \quad \bar{e} = \frac{1}{40} \sum_{i=1}^{40} \left( \frac{1}{5781} \sum_{j=1}^{5781} |e_{ij}| \right) .
\]

(5)

\[
\bar{e}_{large,s} = \frac{\bar{e}_{large} \cdot 100}{K_{rs}} , \quad \text{where} \quad \bar{e}_{large} = \frac{1}{508} \sum_{i=1}^{508} |e_{large}| .
\]

(6)

5. Sensitivity analysis to input elements

The final vector of 13 inputs has been set after an optimization through the sensitivity analysis which is hereafter briefly summarized and discussed in Section 5.1. The number and the type of input elements involved in the input vector for each combination—and the corresponding errors associated—are provided in Table 6.

The first 6 parameters represent the basic parameters involved in the wave reflection process for the simpler cases (i.e. straight slopes under 2D waves) and therefore they have been calibrated on the databases A, B, C, D, see Table 1. This calibration step is described in Section 5.2.

Sections 5.3 and 5.4 are dedicated to sensitivity tests to represent, respectively, structures with berms or toe protections characterized by non-negligible dimension (database E) and oblique wave attacks (database G).

Section 5.5 presents the inclusion of the directional spreading information in the input vector, in order to make the ANN able to work indifferently with both long-crested and short-crested waves.

Finally, Section 5.6 aims to explain why some other input elements have been excluded from the input vector.

5.1. Synthesis of the sensitivity analysis to input elements

Table 6 presents a summary of several results of simulations carried out by changing in each simulation only one input element within the input vector (described in Section 3.1). The percentage error refers to the final model input vector.

By comparing the values of RMSE and WI, it is clear that the ANN is more sensitive to some parameters rather than to others. First, the

Fig. 7. Left: \( |e| = |K_{rs} - K_{ANN}| \geq 0.15 \) (ordinate) as a function of database test indexes (abscissa); right: \( |e| = |K_{rs} - K_{ANN}| \geq 0.15 \) (ordinate) as a function of \( K_{rs} \) (abscissa).
model depends on cotαd and cot αinct, as it is proved by the increased value of \( \text{rmse} \) (up to 46%), which occurs at the contemporary elimination of both parameters. Secondly, a key role is played by \( \beta \) (with an \( \text{rmse} \) increasing of 19%) for non-perpendicular waves. Essential parameters are then \( \frac{h_t}{L_m} - 1.0_0 \) and \( \gamma_f \) which, when omitted, both produce an increase of \( \text{rmse} \) of about +14%. Particularly interesting is the similar influence of \( \frac{H_{m,0,t}}{L_m} - 1.0_0 \) (+11%), \( D_{500}/H_{m,0,t} \) (+11%) and of Spreading (+8%): the nominal diameter results to be more significant than the relative crest-freeboard \( R_c / L_{m,0,t} \) (8%), and the wave directional spreading has apparently the same relevance.

Geometrical parameters, such as \( B/L_m - 1.0_0 \), \( h_t/H_{m,0} \), \( G_t/L_m - 1.0_0 \) and \( m \), seem to be less important if considered separately (each of them providing +5% increase of \( \text{rmse} \)), but together contribute to a proper description of non-straight slopes.

The inclusion of additional parameters also describing toes or crown seaways (e.g., \( h_t/H_{m,0} \) and \( A_{t}/H_{m,0} \)) causes an evident over-fitting and a deterioration of the ANN performance.

### 5.2. Representation of structures with straight slopes

The analysis here discussed refers to the datasets A, B, C and D (Table 1) and to the parameters \( H_{m,0,t}/L_m - 1.0_0 \), \( h_t/L_m - 1.0_0 \), \( R_c/H_{m,0,t} \), cot αd, \( \gamma_f \) and \( D_{500}/H_{m,0,t} \).

The use of the input elements \( H_{m,0,t}/L_m - 1.0_0 \), cot αd, \( \gamma_f \) and \( R_c/H_{m,0,t} \) immediately induces a great response of the network, whereas the inclusion of \( h_t/L_m - 1.0_0 \) and \( D_{500}/H_{m,0,t} \) need a more in depth analysis.

Some interpretation of \( D_{500}/H_{m,0,t} \) is required in case of smooth structures. The final outcome has been artificially set \( D_{500} = 0 \), in order to describe a smooth and impermeable slope.

The parameter \( h_t/L_m - 1.0_0 \) has been compared with the following other formulations:

a. \( h_t/L_m - 1.0_0 \);  
b. \( h_t/H_{m,0,t} \);  
c. \( H_{m,0,t}/h_t \);  
d. elimination of the element from the input vector;  
e. \( h_t/L_m - 1.0_0 \) (as definitely chosen for the optimized model);

The quantitative comparison is completely provided in Table 6. The exclusion of \( h_t/L_m - 1.0_0 \) (case ‘d’) induces an increase of \( \text{rmse} \) slightly lower than 15% (Table 6), being \( \text{rmse} \) on an average equal to 0.042 (note that \( \text{rmse} = 0.038 \) for the optimized model). Its importance is therefore paramount as it is also proved by a larger data scatter and a slight widening of confidence intervals (especially visible in plot ‘d’ in Fig. 8).

Once the introduction of \( h_t \) has been proved essential for the ANN performance, the selection among the alternatives ‘a’, ‘b’, ‘c’ and the final ‘e’ is performed, in order to obtain lower error values and to represent the physical process, i.e. the shoaling parameter.

Fig. 8 presents the comparison among the ANN predicted values, \( K_{e,ANN} \), and the experimental corresponding ones, \( K_{e,a} \), just for cases ‘a’ and ‘d’. These cases have been chosen among the others since resulting qualitatively the most significative, in comparison to the optimal case ‘e’ (the corresponding diagram refers to Fig. 5): case ‘a’ (Fig. 8, to the left) shows an increased scatter, while in case ‘d’ (Fig. 8 to the right) the widening of the confidence interval bands is evident.

### 5.3. Representation of structures with non-straight slopes

To simulate the structures with berms, toe protections and foreshores, it is necessary to introduce some other specific parameters, such as \( G_t/L_m - 1.0_0 \), \( B/L_m - 1.0_0 \), \( h_t/H_{m,0,t} \) and \( m \). The calibration of these parameters has been, at first, carried out by training the network on the datasets E and F (see Table 1) instead of the complete database. It is worthy to note that the two datasets consists of 717 data overall, a number which is just slightly above the 520 connections among the 13 input elements and the 40 hidden neurons. By analogy with structures with straight slopes (Section 5.2), just one of all the tests is here presented, i.e. the choice of \( h_t/H_{m,0,t} \) instead of \( h_t/L_m - 1.0_0 \).

Contrary to the case of \( h_t/L_m - 1.0_0 \) (chosen in place of \( h_t/H_{m,0,t} \)), \( h_t/H_{m,0,t} \) is more suitable and experimentally satisfactory than \( h_t/L_m - 1.0_0 \) to reproduce the berm submergence (i.e. wave breaking on the berm). Indeed, the first one is more sensitive to the increase of the number of hidden neurons, in addition to the better performance results: \( \text{rmse} \)
decreases of 3% and the 95% confidence intervals are therefore narrower, as detectable by comparing Fig. 5 with Fig. 9. Moreover, the ANN with $h_b/L_m - 1.0, t$ tends to under-estimate the actual values of $K_r$, as it is shown by the larger number of scattered points under the continuous bisector line in Fig. 9. Finally, the sensitivity of both parameters to the hidden neurons is shown in Figs. 10 and 11: they respectively represent the values of $\text{rmse}$ and $W_I$ as functions of the increasing number of hidden neurons, respectively. Each of the values refers to the average of the 40 simulations performed.

The values of the standard deviation, as the values of $\text{rmse}$ themselves, are generally larger for the parameter $h_b/L_m - 1.0, t$ (values in Table 6). The better performance of the ANN with $h_b/H_m,0, t$ can be also appreciated by comparing the best fitting curves of $\text{rmse}$ and $W_I$ in Figs. 10 and 11, where the curves for $h_b/L_m - 1.0, t$ are above and below the curves fitting $h_b/H_m,0, t$, respectively. Besides, the element $h_b/H_m,0, t$ clearly shows a greater sensitivity to the increasing dimension of the hidden layer: the ANN over-fitting is evident for a number of hidden neurons greater than 25, since the fitting curves begin to oscillate.

5.4. Representation of oblique wave attacks conditions

The modelling of structures under oblique wave attacks is assigned to the input element $\beta$. In the following, a sample application of the ANN in case of out-of-range values of the relative submergence $K_r/H_m,0, t$ is

![Fig. 8. Comparison among $K_r$ predicted values ($K_{\text{ANN}}$, ordinate) and corresponding $K_r$ experimental values ($K_{r,s}$, abscissa); for cases ‘a’ (inclusion of $h_b/L_m - 1.0, t$ in place of $h_b/L_m - 1.0, t$) and ‘d’ (elimination of $h_b/L_m - 1.0, t$), respectively to the left and to the right.](image)

![Fig. 9. Comparison among $K_r$ predicted values ($K_{\text{ANN}}$, ordinate) and corresponding $K_r$ experimental values ($K_{r,s}$, abscissa); network trained with $h_b/L_m - 1.0, t$ in place of $h_b/L_m - 1.0, t$.](image)

![Fig. 10. $\text{rmse}$ (ordinate) as a function of the number of hidden neurons (abscissa); the square points are associated to the network trained with $h_b/L_m - 1.0, t$ and the triangle points to the network with $h_b/H_m,0, t$. Average results obtained from 40 simulations of the ANN.](image)
performed for a fixed value of $\beta \neq 0$. For oblique cases, i.e. the dataset $G$, $R_c/H_{m0}\theta$ is in the range $[-0.66; 1.90]$ (see Table 3). However, in the full database $R_c/H_{m0}\theta \equiv [-8.10; 25.40]$ It is therefore important to understand if the network can well-estimate $K_c$ in oblique conditions for the whole range of $R_c/H_{m0}\theta$. Simulations have been carried out with artificial data built up so that all the input elements maintained constant values (summarized in Table 7), with the exception of $R_c/H_{m0}\theta$ whose range is between $[-2.00$ and $10.00]$. Since the whole training set is constant, $K_c$ simply becomes a function of $R_c/H_{m0}\theta$, and therefore it is expected to increase as $R_c/H_{m0}\theta$ increases as well. In other terms, it is possible to predict the values of $K_c$ and notice if the network behaves as expected or not. The results of the sensitivity test are shown in Fig. 12: $K_{C,ANN}$ monotonically increases with $R_c/H_{m0}\theta$ but it is upper limited (i.e. $K_{c,ANN} \leq 1$), as it is remarked by the continuous fitting line. This allows to conclude that the ANN seems able to predict $K_c$ in oblique wave attacks even out of the range of training values of $R_c/H_{m0}\theta$. The complete training database is evidently wide enough to make the ANN providing correct predictions also in pretty various conditions.

5.5. Wave directional spreading

The need of including an input element such as the wave directional spreading has been investigated thoroughly, by analyzing the predictions of the ANN in case of oblique and 3D wave attacks.

Table 7

<table>
<thead>
<tr>
<th>element</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$H_{m0}\theta/H_m - 1.0%$</td>
<td>0.076</td>
</tr>
<tr>
<td>$b/L_m - 1.0%$</td>
<td>0.170</td>
</tr>
<tr>
<td>$R_c/H_{m0}\theta$</td>
<td>$[-2.00; +10.00]$</td>
</tr>
<tr>
<td>$\cot \alpha_c$</td>
<td>0.962</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>2.000</td>
</tr>
<tr>
<td>$D_{x0}/H_{m0}\theta$</td>
<td>0.385</td>
</tr>
<tr>
<td>$c_c/L_m - 1.0%$</td>
<td>0.046</td>
</tr>
<tr>
<td>$\delta$</td>
<td>60.000</td>
</tr>
<tr>
<td>$b/L_m - 1.0%$</td>
<td>0.000</td>
</tr>
<tr>
<td>$h_B/H_{m0}\theta$</td>
<td>0.000</td>
</tr>
<tr>
<td>$\cot \alpha_{h_B}$</td>
<td>2.000</td>
</tr>
<tr>
<td>$m$</td>
<td>1000.00</td>
</tr>
<tr>
<td>spreading</td>
<td>5.000</td>
</tr>
</tbody>
</table>

Taking into account the wave directional spreading allows to detect the different behavior of short-crested and long-crested waves, therefore, it leads to an improvement of the ANN performance (rmse decreases of 8%).

Fig. 13 shows the ANN performance by eliminating and adding the spreading parameter. The comparison among the values of $K_{c,ANN}$ and $K_c$ is presented just for oblique wave attacks distinguishing among short and long-crested waves. The plot to the left of Fig. 13 clearly denotes that the ANN is not able to deal with short-crested waves (i.e. spreading $\neq 0$) if the wave directional spreading information is missing: with the exception of a few cases, all the $K_{c,ANN}$ values are systematically underestimated, as it is demonstrated by the large data scatter under the continuous perfect fit line. At the contrary, the inclusion of the spreading (Fig. 13 to the right) solves most of the shortcomings related to short-crested waves and also improves the interpretation of long-crested waves. Moreover, it permits to reduce the width of the confidence intervals.

5.6. Non-significant parameters

Since Fig. 7 shows some systematic errors associated to the same tests, it is clear that the final ANN still presents some difficulties in modelling particular structures. The diagram to the left denotes frequent recurrent errors for the following indexes: $[3000; 3500]$ (smooth and impermeable slopes), $[4500; 5400]$ (structures with non-straight slopes) and $>5400$ (vertical seawalls), as remarked already in Section 4.3.

To eliminate or, at least, minimize these residual effects, some other parameters have been conceived in order to possibly provide a better reproduction of wave breaking also in presence of toes and berms. The parameters which have been tested are listed in the following:

a. $h_B/H_{m0}$ to model the wave breaking over the toe;
b. $h_B/h$ to represent the degree of depth-induced breaking due to the difference between the depth at the toe and over the toe;
c. $h/H_{m0}$, already-tested input element as alternative to $h/L_m - 1.0\%$, but hereafter considered as an additional information to evaluate the breaking index.

Fig. 14 displays the "large" errors $e > |0.15|$ computed by the ANN as functions of the data indexes for case ‘a’. This kind of plot, which provides an example of the ANN answer to the introduction of one additional parameter in the input set, is suitable for a direct comparison with Fig. 7 (left). The diagrams of Fig. 14 shows not only an increased number of "large" errors, but also increased values of the errors themselves (for cases ‘b’ and ‘c’—not displayed here—the quantity $e$ is more frequently greater than 0.4).

The quantitative analysis—which confirms the qualitative results—is again provided in Table 6 where both values of rmse and WI worsen. It can therefore be concluded that the input elements ‘a’, ‘b’ and ‘c’ induce a certain over-fitting disease to the ANN.

The crown seawall modelling is finally studied. This element can produce further reflection over the structure crest or at its off-shore edge, especially when it is particularly high and the crest is sufficiently narrow. Moreover, it should be considered as an additional element that alters the average value of cot $\alpha_c$. Looking at Fig. 1, the only option to represent the difference of total structure height above sea level in absence/presence of a crown seawall consists in adding the relative crest height $A_c$ to the selected 13 input elements. Being $A_c$ a vertical element, it can be naturally non-dimensionalized through the wave height $H_{m0}\theta$. Similarly to the toe protection parameters, also $A_c/H_{m0}\theta$, as 14th input element, provides a slightly worse performance (see Table 6) and generates more "large" errors. This is the reason why not even $A_c/H_{m0}\theta$ has been introduced in the final input vector.
6. Sensitivity analysis to neural network structures and algorithms

Besides establishing the best input vector configuration, the process of model optimization includes also a calibration phase to well define the best algorithm characteristics of a neural network model (see the ANN architecture in Section 3.1). The most relevant structures to be calibrated for the training step are undoubtedly the definition of the dimensions of the hidden layer and the designation of the training numerical algorithm. The number of hidden neurons has been finally set to 40 (see Section 6.1) and the Levenberg–Marquardt training algorithm (Levenberg, 1944 and Marquardt, 1963) has been chosen (see Section 6.2).

6.1. Number of hidden neurons

The ANN should reach a compromise to satisfy the necessity to learn and reproduce accurately the input–output relations as well as the necessity to prevent over-fitting risk. This compromise can be translated
into a hidden layer dimension compromise: the most suitable number of hidden neurons can be fixed by testing and simulating the network more times, using each time an increasing number of neurons.

The typical trend of rmse and WI as functions of the number of hidden neurons is reported in Figs. 15 and 16: the values of the indexes respectively decrease and increase until the network runs up against the over-fitting.

From both the diagrams it is clear that the ideal number of hidden neurons is 40, since this value corresponds both to a minimum of rmse and a maximum of WI, as it is remarked by the trend of the fitting lines. The standard deviation is minimized in correspondence of a higher number of hidden neurons (45), but this modest improvement would not justify the increased dimensions of the hidden layer and therefore an increased number of parameters. As always, both the graphs refer to average values, obtained from 40 simulations of the ANN.

6.2. Training algorithm

The best training function for the ANN turned out to be the Levenberg–Marquardt algorithm (Levenberg, 1944 and Marquardt, 1963). It can develop a second order computing efficiency without calculating the Hessian matrix and so limiting the computational cost.

The calibration of the training algorithm should be associated to the calibration of the learning algorithm. Training and learning algorithms are dozens and the combined sensitivity analysis is very complex. Hereafter only the most common training-learning algorithms are listed:

a. gradient descent (“gd”) algorithm, basic formulation;
b. gradient descent (“gd”) algorithm modified with an adaptive learning rate; it allows the learning rate to vary according to the error function trend;
c. resilient back-propagation algorithm (Reidmiller, 1993); it requires neurons transfer functions differentiability and it is generally faster than the gradient descent algorithms, by involving just a little increment of computational memory;
d. scaled conjugate gradient algorithm (Møller, 1993); it also requires neuron transfer functions differentiability.

The average performance of these algorithms is reported in Table 8, in comparison to the Levenberg–Marquardt’s. With the exception of the training algorithm, the whole ANN architecture is the same as described in Section 3.1.

The 4 algorithms provide gradually a better performance, but only the last two ones (resilient back-propagation and scaled conjugate gradient algorithms) are characterized by errors at least comparable to the Levenberg–Marquardt ones. The adaptive learning rate (case ‘b’) induces some improvement with respect to the simple gradient descent algorithm (case ‘a’), but it is clear that the “gd” algorithms are not suitable for this ANN.

7. Comparison among the new ANN and existing prediction formulae

The aim of the present section is to provide a comparison among ANN performance and traditional prediction formulae. Due to limited ranges of validity associated to existing formulae, the comparison has been carried out against narrower datasets selected among the whole database employed to train the network. Two cases
have been considered: perpendicular wave attacks on straight slopes (described in Section 7.1) and oblique wave attacks (Section 7.2).

7.1. Straight slopes

The ANN performance is compared with the results of Zanuttigh and Van der Meer (2008) formula (Eq. (1), hereafter ZVDM) in case of straight slopes under perpendicular wave attacks. By applying the ZVDM formula ranges of validity (i.e. design condition, $R_c/H_{m0,\infty} \geq 1.0$, $s_0 = H_{m0}/L_m - 1.0, t \geq 0.01$) to the whole database, a total of 724 data was selected (corresponding to part of the datasets A, B, C and D, see Table 1).

Fig. 17 shows the comparison among $K_r$ predicted values from the ANN and $K_r$ experimental values as functions of $K_r$, $ZVDM$, with increasing the relative crest freeboard $R_c/H_{m0,\infty}$. In each diagram, the data have been distinguished according to the type of armor (rock/unit) and core (permeable/impermeable).

Figs. 17 and 18 show that the values obtained from the ANN $K_r, ANN$ are characterized by a much lower dispersion around the ideal condition than the values $K_r, ZVDM$ (Fig. 17 to the left).

Fig. 17. Comparison among $K_r$ predicted values (ordinate) and corresponding $K_r$ experimental values (abscissa) for ZVDM formula predictions (on the left) and ANN predictions (on the right). Structures with straight slopes only considered.

Fig. 18. Comparison among $K_r$ predicted values in proportion to the corresponding $K_r$ experimental ones (ordinate), as functions of the relative crest freeboard $R_c/H_{m0,\infty}$ (abscissa, logarithmic scale) for ZVDM formula predictions (on the left) and ANN predictions (on the right). Structures with straight slopes only considered.
Table 9

rmse, WI and $R^2$ values obtained simulating ANN vs ZVDM formula (to the left) and ANN vs ZLA formula (to the right). Respectively, structures with straight slopes (left) and oblique wave attacks (right) only considered.

<table>
<thead>
<tr>
<th></th>
<th>Straight slopes</th>
<th>Oblique wave attacks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ZVDM</td>
<td>ANN</td>
</tr>
<tr>
<td>rmse</td>
<td>0.041</td>
<td>0.027</td>
</tr>
<tr>
<td>WI</td>
<td>0.925</td>
<td>0.973</td>
</tr>
<tr>
<td>$R^2$</td>
<td>0.75</td>
<td>0.89</td>
</tr>
</tbody>
</table>

Furthermore, the ANN predictions seem not affected by:

- the armor/core type, while ZVDM formula tends to systematically over-estimate $K_r$ for rocks and permeable slopes and under-estimate $K_r$ for rocks over impermeable core (Fig. 17);
- the relative crest freeboard, while ZVDM formula tends to systematically over-estimate $K_r$ for $R_c/H_m,0.5 < 2$ and to underestimate for $R_c/H_m,0.5 > 3$ (Fig. 18).

Table 9 (to the left) reports the quantitative results of these simulations in terms of rmse, WI and $R^2$ values. The ANN allows for more accurate predictions even considering the same range of data the ZVDM formula was based on.

7.2. Oblique wave attack conditions

In case of oblique wave attacks, the ANN performance is compared with the following formula by Zanuttigh and Lykke Andersen (2010), ZLA hereafter:

$$K_{r ZLA}(\beta, s) = K_{r ZVDM}(\beta = 0, s) \cdot \gamma_{\beta, s},$$

where: $\gamma_{\beta, s} = \begin{cases} (1 - 0.0037) \cdot \beta, & \text{for long-crested waves} \\ (1 - 0.0058) \cdot \beta, & \text{for short-crested waves}. \end{cases}$ (7)

The expression of the wave obliquity factor $\gamma_{\beta, s}$ in Eq. (7) is provided by Lykke Andersen and Burcharth (2009). The wave reflection coefficient $K_r ZVDM$ is still given by Eq. (1) as above.

The comparison is performed over 736 tests throughout the dataset $G$ (see Table 1) fulfilling the design conditions associated to ZVDM and ZLA formulae.

The comparisons among rmse, WI and $R^2$ values are still reported in Table 9, to the right. Again, as in Section 7.1, the ANN provides more accurate predictions even considering the same range of data the ZLA formula was based on.

A qualitative analysis is instead provided by the diagrams of Figs. 19 and 20. The predicted values of $K_r$ by the ANN and the ZLA formula are compared against experimental values in Fig. 19; the dispersion of relative values $K_{r ZLA}/K_{r s}$ and $K_{r ANN}/K_{r s}$ are shown as functions of $\beta$ in Fig. 20. In these diagrams short-crested and long-crested waves are also distinguished.

The ZLA formula tends to perform greater errors for greater values of $\beta$ (Fig. 20), while the ANN seems to be not affected by $\beta$. Both the ZLA formula and the ANN do not show any particular sensitivity to wave directional spreading (Fig. 19).

8. Uncertainty of the prediction

The uncertainty of the prediction of the proposed ANN has been assessed following the CLASH approach presented in Van Gent et al. (2007), which is based on a bootstrapping resampling technique. A number of bootstrap resamples are generated starting from the original database. Each of these resampled databases has the same number of samples of the original one and is randomly drawn from it. Replacements are allowed, so that each sample of the original set can either appear more than once or not at all in each of the bootstrap resamples.

Each of these sets is used for training and testing/validating the ANN here described. Following Van Gent et al. (2007), 500 databases are resampled and the corresponding ANNs are tested. Each simulation provides the $j$-th $(j = 1, 2, \ldots, 5781$ – denoting the test number) estimate for the $i$-th $(i = 1, 2, \ldots, 500 –$ denoting the resampling number) resample, i.e. $(K_{r ANN})_i$, allowing to obtain the mean estimate for the reflection coefficient:

$$K_{r ANN} = \frac{1}{L} \sum_{i=1}^{L} (K_{r ANN})_i,$$

where $L = 500$.  

![Fig. 19. Comparison among $K_r$ predicted values (ordinate) and corresponding $K_r$ experimental values (abscissa) for ZLA formula predictions (on the left) and ANN predictions (on the right). Oblique wave attacks only considered.](image-url)
Van Gent et al. (2007) proposed to modify the cost function used in the ANN training to weight each test differently. This is done by using a weighting factor (WF) defined as

\[ WF = (4 - RF)(4 - CF), \]  

where RF and CF are respectively the reliability and the complexity factors. It is RF = 1 for very reliable data and RF = 4 for unreliable ones. CF = 1 indicates a simple structure that is completely described by the geometric parameters used, while CF = 4 indicates structures whose cross-section is so complex that the adopted parameters may not describe it accurately. The criteria used to assign RF are the same as those used in CLASH project and reported in Van Gent et al. (2007).

Table 10 describes the distribution of RF and CF in the database. Less than 10% of the total database shows RF > 2 and only the 27% of the database shows CF > 2. The WFs are introduced in the error function used in the training of the network in the expression of the j-th error in each of the i-th resamples:

\[ WF \cdot \left( K_{r,s} - K_{r,\text{ANN}} \right)^2. \]  

(10)

Note that, given the structure of WF, samples with RF or CF equal to 4 will not contribute to the training of the ANN.

Fig. 21 shows the comparison among \( K_{r,\text{ANN}} \) and \( K_{r,s} \). The overall performance is given in Table 11. The non-significant difference of the ANN performance with and without bootstrapping can be explained with the abundance of data characterized by high WF and may be regarded as a confirmation of the ANN stability.

9. Conclusions

An ANN for the prediction of the wave reflection coefficient at coastal structures has been developed. It has been trained on a wide database (5781 data) including structures with straight and non-straight slopes; seawalls, caissons and circular caissons; Acquareefs and structures under oblique wave attacks.

The ANN is characterized by 13 input elements and 40 hidden neurons. The input elements have been selected to include the most significant wave and structure characteristics: \( H_{m,0}, L_m - 1.0, h_t \), \( L_m - 1.0, R_c/H_{m,0}, \cot \alpha_s \), \( g_j, D_{j,50}/H_{m,0}, \beta, G_c, L_m - 1.0, B/L_m - 1.0, h_t/H_{m,0}, \cot \alpha_{inr}, m, spreading. \) An in-depth sensitivity analysis has been performed for each of these parameters, in order to define the best input pattern, while the optimal number of 40 hidden neurons has been identified as a compromise between a satisfactory performance and the prevention of the risk of over-fitting.

This ANN is able to provide accurate predictions \( K_{r,\text{ANN}} \) of the measured wave reflection coefficient \( K_{r,s} \), being on average characterized by \( \text{rmse} \approx 0.038 \) and \( \text{Wt} \approx 0.985 \); the values of the standard deviation are \( \approx 10^{-3} \) for both the indexes.

Table 10

<table>
<thead>
<tr>
<th>Value</th>
<th>RF, #</th>
<th>RF, %</th>
<th>CF, #</th>
<th>CF, %</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2008</td>
<td>34.7%</td>
<td>3888</td>
<td>67.2%</td>
</tr>
<tr>
<td>2</td>
<td>3212</td>
<td>55.6%</td>
<td>1258</td>
<td>5.6%</td>
</tr>
<tr>
<td>3</td>
<td>398</td>
<td>6.7%</td>
<td>1408</td>
<td>24.4%</td>
</tr>
<tr>
<td>4</td>
<td>171</td>
<td>3.0%</td>
<td>160</td>
<td>2.8%</td>
</tr>
</tbody>
</table>

Fig. 21. Comparison between \( K_r \) predicted values (\( K_{r,\text{ANN}} \), ordinate) and corresponding \( K_r \) experimental values \( K_{r,s} \) (abscissa). Bootstrap resampling results.
Table 11
rmse and WI average values and corresponding standard deviation provided by 500 bootstrap resamples of the original database.

<table>
<thead>
<tr>
<th></th>
<th>Mean</th>
<th>Stand. Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>rmse</td>
<td>0.037</td>
<td>0.003</td>
</tr>
<tr>
<td>WI</td>
<td>0.985</td>
<td>0.003</td>
</tr>
<tr>
<td>$R^2$</td>
<td>0.96</td>
<td>0.02</td>
</tr>
</tbody>
</table>

Some systematic “large” errors (i.e. difference $K_{cs} - K_{cs,ANN} \geq 0.15$), are typically associated to non-straight and smooth slopes. In the worst case there is a 0.2% frequency of relative errors greater than 80%. These results appear pretty satisfactory, especially if considering the great variety of data types which the ANN deals with.

Besides, the predictions of $K_r$ provided by the ANN are more accurate than existing empirical formulae, if applied to the same datasets employed to derive the formulae themselves. For instance, for straight slopes under perpendicular wave attacks rmse equals 0.027 and 0.041 if one uses the ANN or Zanuttigh and Van der Meer (2008) formula, respectively.

Further research should be performed to address the accurate representation of more complex geometries in the ANN, such as breakwaters or vertical walls with double crown walls or combined crown walls (i.e. crown walls composed by different slopes), breakwaters with storage basins on the crest, perforated and Jarlan-type caisson. The modelling of the wave reflection resulting from the combination of two structures has still to be examined; for instance, in a changing climate, the practical case of the upgrade of an existing dike by placing a submerged porous structure in front it may be of particular interest.

List of notations

ANN(s) Acronym of Artificial Neural Network(s)

$\alpha$ Coefficient depending on $\gamma_f$ in ZVDM formula, see Eq. (1)

$A_c$ Armor crest freeboard of the structure

$B$ Width of the berm

$B_h$ Width of the horizontally schematized berm

$B_t$ Toe width

$b$ Coefficient depending on $\gamma_f$ in ZVDM formula, see Eq. (1)

$b^*$ Bias, see Fig. 3

$CF$ Complexity factor, describing the degree of complexity of the geometry of a structure in the database

$D_{n,SO}$ Nominal rock diameter or typical armor unit size

$e$ Difference between the experimental wave reflection coefficient and the predicted wave reflection coefficient

$G_c$ Structure crest width

$g$ Acceleration due to gravity

$H_{m0,deep}$ Significant incident wave height at deep water

$H_{m0,t}$ Significant incident wave height at the structure toe

$h$ Water depth at the structure toe

$h_b$ Berm submergence

$h_{deep}$ Water depth at deep water

$h_t$ Water depth above the structure toe

$m$ Foreshore slope, defined as the cotangent of the angle of the slope itself

$m_n$ $n$-th moment of spectral density

$\text{rmse}$ Acronym of Root Mean Squared Error

$\text{SWL}$ Acronym of Still Water Level

$\alpha_{incl}$ Mean slope of the structure downward of the berm

$\alpha_{excl}$ Mean slope of the structure within the run-up and run-down zone, excluding the berm; the mean angle is defined between $\pm 1.5H_{m0,t}$

$\alpha_t$ Toe slope

$\alpha_s$ Slope of the structure upward of the berm

$\beta$ Angle of deviation from the perpendicular wave attack direction

$\gamma_f$ Roughness factor as found in overtopping research

$\xi_0$ Breaker parameter based on spectral wave period at the structure toe

Acknowledgments

The support of the European Commission through FP7.2009-1, Contract 244104—THESEUS project ("Innovative technologies for safer European coasts in a changing climate"), www.theseusproject.eu, is gratefully acknowledged.

The first author wishes to express her gratitude to Dr. Jentsje W. van der Meer for the helpful data handling ad kind long-term cooperation.

Dr. Andreas Kortenhaus is acknowledged for providing additional datasets on vertical walls and mobile flood defence systems.

The third author wishes to express his gratitude to the EPSRC for supporting him through a Career Acceleration Fellowship (EP/1004505/1).

Last but not least, the Authors would like to gratefully acknowledge two anonymous reviewers who contributed to a significant improvement of this paper.
### Table 12
Weight matrices of the connections; from left to right: IW(1,1), size (40 × 40); LW(2,1), size (1 × 40); b^1(1,1), size (40 × 1); b^2(1,1), size (1 × 1).

<table>
<thead>
<tr>
<th>IW(1,1), size 40 × 40</th>
<th>LW(2,1), size 1 × 40</th>
<th>b^1(1,1), size 40 × 1</th>
<th>b^2(1,1), size 1 × 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.7441</td>
<td>−3.3135</td>
<td>0.2910</td>
<td>−2.7538</td>
</tr>
<tr>
<td>1.3164</td>
<td>11.1142</td>
<td>−1.8315</td>
<td>4.9742</td>
</tr>
<tr>
<td>−4.1938</td>
<td>−7.1516</td>
<td>0.9015</td>
<td>2.8941</td>
</tr>
<tr>
<td>−1.4048</td>
<td>1.6335</td>
<td>−3.4214</td>
<td>1.7761</td>
</tr>
<tr>
<td>−0.2839</td>
<td>0.6744</td>
<td>1.4775</td>
<td>−1.8685</td>
</tr>
<tr>
<td>−1.0820</td>
<td>10.9598</td>
<td>−3.9681</td>
<td>−4.1277</td>
</tr>
<tr>
<td>0.0575</td>
<td>1.0607</td>
<td>4.1989</td>
<td>4.4907</td>
</tr>
<tr>
<td>−1.7574</td>
<td>0.9184</td>
<td>0.6862</td>
<td>2.2344</td>
</tr>
<tr>
<td>−1.3246</td>
<td>−1.7513</td>
<td>7.5621</td>
<td>−5.0428</td>
</tr>
<tr>
<td>−0.1956</td>
<td>−4.3331</td>
<td>4.6659</td>
<td>0.0987</td>
</tr>
<tr>
<td>4.4841</td>
<td>−5.5521</td>
<td>6.4965</td>
<td>2.0813</td>
</tr>
<tr>
<td>0.6238</td>
<td>−3.3094</td>
<td>2.8785</td>
<td>3.0556</td>
</tr>
<tr>
<td>−0.1602</td>
<td>2.6911</td>
<td>−0.7427</td>
<td>1.9393</td>
</tr>
<tr>
<td>−0.0149</td>
<td>−9.8980</td>
<td>−12.5007</td>
<td>−4.1742</td>
</tr>
<tr>
<td>−1.3750</td>
<td>−7.2163</td>
<td>−2.2806</td>
<td>5.1944</td>
</tr>
<tr>
<td>−0.0384</td>
<td>0.5675</td>
<td>−2.1741</td>
<td>−0.9861</td>
</tr>
<tr>
<td>0.3046</td>
<td>−5.5590</td>
<td>9.8650</td>
<td>−2.2027</td>
</tr>
<tr>
<td>−0.0457</td>
<td>3.1127</td>
<td>0.9173</td>
<td>−8.9877</td>
</tr>
<tr>
<td>−0.0842</td>
<td>0.3063</td>
<td>−1.1821</td>
<td>1.1642</td>
</tr>
<tr>
<td>−1.8065</td>
<td>−7.9673</td>
<td>1.6939</td>
<td>−5.0415</td>
</tr>
<tr>
<td>2.0689</td>
<td>6.8918</td>
<td>−4.1516</td>
<td>3.2846</td>
</tr>
<tr>
<td>3.3283</td>
<td>3.5747</td>
<td>3.1324</td>
<td>−0.3254</td>
</tr>
<tr>
<td>0.5986</td>
<td>−7.3867</td>
<td>−6.1913</td>
<td>−2.6671</td>
</tr>
<tr>
<td>7.5970</td>
<td>−2.9890</td>
<td>20.8853</td>
<td>−6.1960</td>
</tr>
<tr>
<td>−2.9121</td>
<td>−2.9257</td>
<td>−1.4794</td>
<td>−0.8414</td>
</tr>
<tr>
<td>1.9631</td>
<td>−6.4503</td>
<td>−5.3814</td>
<td>3.7733</td>
</tr>
<tr>
<td>−0.9715</td>
<td>−1.1326</td>
<td>3.7148</td>
<td>−3.1263</td>
</tr>
<tr>
<td>2.4110</td>
<td>4.1605</td>
<td>0.1496</td>
<td>−2.1669</td>
</tr>
<tr>
<td>6.3481</td>
<td>0.2416</td>
<td>−12.0944</td>
<td>12.9463</td>
</tr>
<tr>
<td>−0.8450</td>
<td>−2.6156</td>
<td>4.2199</td>
<td>1.1694</td>
</tr>
<tr>
<td>2.0708</td>
<td>−3.7458</td>
<td>5.1875</td>
<td>−1.0611</td>
</tr>
<tr>
<td>1.2050</td>
<td>−3.6909</td>
<td>−0.9754</td>
<td>5.3492</td>
</tr>
<tr>
<td>0.5639</td>
<td>−3.9866</td>
<td>−0.1915</td>
<td>5.9077</td>
</tr>
<tr>
<td>−0.9794</td>
<td>22.2359</td>
<td>−3.6023</td>
<td>−1.0767</td>
</tr>
<tr>
<td>−1.6155</td>
<td>14.0942</td>
<td>2.0490</td>
<td>0.2218</td>
</tr>
<tr>
<td>−0.9825</td>
<td>2.8766</td>
<td>−1.6968</td>
<td>−6.9039</td>
</tr>
<tr>
<td>−9.6043</td>
<td>4.0313</td>
<td>−1.5780</td>
<td>−10.9126</td>
</tr>
<tr>
<td>−2.5771</td>
<td>7.7043</td>
<td>−1.5896</td>
<td>−6.0610</td>
</tr>
<tr>
<td>−2.4803</td>
<td>7.5752</td>
<td>−1.8069</td>
<td>5.0194</td>
</tr>
</tbody>
</table>

b^2(1,1), size 1 × 1
4.3538