1 $PLS-OSF-SVR$ Approach

GOAL: The main goal is to jointly minimize

- the number N of training samples (i.e., the number of simulations / the computational time required to generate the training set);
- the prediction error made by the LBE technique in estimating the crack parameters.

1.1 Algorithm #1

1. **Partial Least Squares (PLS)**

   (a) Build an initial set $\mathbf{P}_{N_1}$ of N_1 configurations of the crack (dim. $[N_1 \times I]$) (e.g., positions within the plate)

   $$\mathbf{P}_{N_1} = \begin{bmatrix} \mathbf{p}^{(1)} \\ \vdots \\ \mathbf{p}^{(N_1)} \end{bmatrix} = \begin{bmatrix} p_1^{(1)} & \dots & p_I^{(1)} \\ \vdots & \ddots & \vdots \\ p_1^{(N_1)} & \dots & p_I^{(N_1)} \end{bmatrix} = \begin{bmatrix} x_0^{(1)} & y_0^{(1)} & z_0^{(1)} \\ \vdots & \vdots & \vdots \\ x_0^{(N_1)} & y_0^{(N_1)} & z_0^{(N_1)} \end{bmatrix}$$

   using a uniform grid sampling in the I-dimensional space of crack parameters;

   (b) Use the forward solver $\Phi\{.\}$ to compute the ECT signal in K measurement points associated to the N_1 configurations of the crack. Build the following matrix of measurements where real and imaginary parts of each measurement point are treated as separate real-valued features (dim. $[N_1 \times 2K] = [N_1 \times F]$)

   $$\mathbf{\Psi}_{N_1} = \begin{bmatrix} \mathbf{\Psi}^{(1)} \\ \vdots \\ \mathbf{\Psi}^{(N_1)} \end{bmatrix} = \begin{bmatrix} \Re\left\{\Psi_1\left(\mathbf{p}^{(1)}\right)\right\} & \Im\left\{\Psi_1\left(\mathbf{p}^{(1)}\right)\right\} & \dots & \Re\left\{\Psi_K\left(\mathbf{p}^{(1)}\right)\right\} & \Im\left\{\Psi_K\left(\mathbf{p}^{(1)}\right)\right\} \\ \vdots & & \ddots & & \vdots \\ \Re\left\{\Psi_1\left(\mathbf{p}^{(N_1)}\right)\right\} & \Im\left\{\Psi_1\left(\mathbf{p}^{(N_1)}\right)\right\} & \dots & \Re\left\{\Psi_K\left(\mathbf{p}^{(N_1)}\right)\right\} & \Im\left\{\Psi_K\left(\mathbf{p}^{(N_1)}\right)\right\} \end{bmatrix};$$

   (c) Use PLS to extract from matrix $\mathbf{\Psi}_{N_1}$ $J << F$ features to be used during the inversion. In this version of the algorithm, each i-th parameter to estimate is treated in a separate way. For each parameter ($i = 1, ..., I$) the PLS algorithm receives as input the pair

   $$\{\mathbf{\Psi}_{N_1}; \mathbf{P}_{N_1,i}\}, \quad i = 1, ..., I$$

   where $\mathbf{P}_{N_1,i}$ is the i-th column of $\mathbf{P}_{N_1}$(i.e., the value of the i-th parameter of the crack for the N_1 configurations)

   $$\mathbf{P}_{N_1,i} = \begin{bmatrix} p_i^{(1)} \\ \vdots \\ p_i^{(N_1)} \end{bmatrix}, \quad i = 1, ..., I$$

ELEDIA Research Center

and retrieves a matrix $\mathbf{T}_{N_1,i}$ of extracted J-dimensional features (dim. $[N_1 \times J]$) that are the most informative ones for predicting $\mathbf{P}_{N_1,i}$

$$\mathbf{T}_{N_1,i} = \begin{bmatrix} \mathbf{t}_i^{(1)} \\ \vdots \\ \mathbf{t}_i^{(N_1)} \end{bmatrix} = \begin{bmatrix} t_{i,1}^{(1)} & ... & t_{i,J}^{(1)} \\ \vdots & \ddots & \vdots \\ t_{i,1}^{(N_1)} & ... & t_{i,J}^{(N_1)} \end{bmatrix}, \quad i = 1, ..., I$$

as well as the matrix $\mathbf{W}_i$ of weights (dim. $[F \times J]$) used to perform a linear projection of $\mathbf{\Psi}_{N_1}$ into $\mathbf{T}_{N_1,i}$ (i.e., compress the original $F = 2K$ features to the new extracted J features)

$$\mathbf{T}_{N_1,i} = \left(\mathbf{\Psi}_{N_1} - \overline{\mathbf{\Psi}_{N_1}}\right) \times \mathbf{W}_i$$

where $\overline{\mathbf{\Psi}_{N_1}}$ is a matrix containing the average values of the columns of $\mathbf{\Psi}_{N_1}$

$$\overline{\mathbf{\Psi}_{N_1}} = \begin{bmatrix} \overline{\Re\{\Psi_1\}} & \overline{\Im\{\Psi_1\}} & ... & \overline{\Re\{\Psi_K\}} & \overline{\Im\{\Psi_K\}} \\ \vdots & & \ddots & & \vdots \\ \overline{\Re\{\Psi_1\}} & \overline{\Im\{\Psi_1\}} & ... & \overline{\Re\{\Psi_K\}} & \overline{\Im\{\Psi_K\}} \end{bmatrix};$$

and

$$\begin{cases} \overline{\Re\{\Psi_k\}} = \frac{1}{N_1} \sum_{n=1}^{N_1} \Re\left\{\Psi_k\left(\mathbf{p}^{(n)}\right)\right\} \\ \overline{\Im\{\Psi_k\}} = \frac{1}{N_1} \sum_{n=1}^{N_1} \Im\left\{\Psi_k\left(\mathbf{p}^{(n)}\right)\right\} \end{cases} k = 1, ..., K$$

(d) The initial training sets of N_1 samples are then composed by the input-output pairs

$$\left\{\mathbf{T}_{N_1,i}; \mathbf{P}_{N_1,i}\right\} = \left\{ \begin{bmatrix} t_{i,1}^{(1)} & ... & t_{i,J}^{(1)} \\ \vdots & \ddots & \vdots \\ t_{i,1}^{(N_1)} & ... & t_{i,J}^{(N_1)} \end{bmatrix}; \begin{bmatrix} p_i^{(1)} \\ \vdots \\ p_i^{(N_1)} \end{bmatrix} \right\} \quad i = 1, ..., I.$$

2. **Output Space Filling (*OSF*).**

Set $N = N_1$. To generate a new training sample perform the following steps

(a) Generate C "candidate" configurations of the crack, by sampling with LHS the I-dimensional space of crack parameters

$$\mathbf{P}_C = \begin{bmatrix} \mathbf{p}^{(1)} \\ \vdots \\ \mathbf{p}^{(C)} \end{bmatrix} = \begin{bmatrix} p_1^{(1)} & ... & p_I^{(1)} \\ \vdots & \ddots & \vdots \\ p_1^{(C)} & ... & p_I^{(C)} \end{bmatrix} = \begin{bmatrix} x_0^{(1)} & y_0^{(1)} & z_0^{(1)} \\ \vdots & \vdots & \vdots \\ x_0^{(C)} & y_0^{(C)} & z_0^{(C)} \end{bmatrix}$$

(b) Use the forward solver to compute the ECT signal associated to the C candidates. Build the following matrix of measurements (dim. $[C \times 2K] = [C \times F]$)

FLEDIA Research Center

$$\mathbf{\Psi}_C = \begin{bmatrix} \mathbf{\Psi}^{(1)} \\ \vdots \\ \mathbf{\Psi}^{(C)} \end{bmatrix} = \begin{bmatrix} \Re\left\{\Psi_1\left(\mathbf{p}^{(1)}\right)\right\} & \Im\left\{\Psi_1\left(\mathbf{p}^{(1)}\right)\right\} & ... & \Re\left\{\Psi_K\left(\mathbf{p}^{(1)}\right)\right\} & \Im\left\{\Psi_K\left(\mathbf{p}^{(1)}\right)\right\} \\ \vdots & & \ddots & & \vdots \\ \Re\left\{\Psi_1\left(\mathbf{p}^{(C)}\right)\right\} & \Im\left\{\Psi_1\left(\mathbf{p}^{(C)}\right)\right\} & ... & \Re\left\{\Psi_K\left(\mathbf{p}^{(C)}\right)\right\} & \Im\left\{\Psi_K\left(\mathbf{p}^{(C)}\right)\right\} \end{bmatrix}$$

(c) For each i-th parameter of the crack:

i. project the matrix of measurements $\mathbf{\Psi}_C$ onto the reduced J-dimensional space using the i-th transformation matrix $\mathbf{W}_i$

$$\mathbf{T}_{C,i} = \left(\mathbf{\Psi}_C - \overline{\mathbf{\Psi}_{N_1}}\right) \times \mathbf{W}_i = \begin{bmatrix} \mathbf{t}_i^{(1)} \\ \vdots \\ \mathbf{t}_i^{(C)} \end{bmatrix} = \begin{bmatrix} t_{i,1}^{(1)} & ... & t_{i,J}^{(1)} \\ \vdots & \ddots & \vdots \\ t_{i,1}^{(C)} & ... & t_{i,J}^{(C)} \end{bmatrix}, \quad i = 1, ..., I$$

ii. Compute the Euclidean distance between each c-th vector $\mathbf{t}_i^{(c)} (c = 1, ..., C)$ and the vectors $\mathbf{t}_i^{(n)}, n = 1, ..., N$ previously collected (i.e., the rows of $\mathbf{T}_{N,i}$)

$$dist\left\{\mathbf{t}_i^{(c)}, \mathbf{t}_i^{(n)}\right\} = \sqrt{\sum_{j=1}^{J}\left(t_{i,j}^{(c)} - t_{i,j}^{(n)}\right)^2}, \quad c = 1, ..., C, \quad n = 1, ..., N$$

iii. Select the vector of extracted features $\mathbf{t}_i^{(*)}$ having the maximum minimum distance with respect to the previously collected N vectors $(\mathbf{t}_i^{(n)}, n = 1, ..., N)$

$$\mathbf{t}_i^{(*)} = arg\left\{\max_{c=1,...,C}\left[\min_{n=1,...,N}\left(dist\left\{\mathbf{t}_i^{(c)}, \mathbf{t}_i^{(n)}\right\}\right)\right]\right\}$$

this vector corresponds to the candidate sample $\mathbf{p}^{(*)}$ that is the farthest one in the space of extracted features for the i-th parameter;

iv. Append $\mathbf{t}_i^{(*)}$ and the associated crack parameter $p_i^{(*)}$ to the training set, that will be composed as

$$\{\mathbf{T}_{N+1,i}; \mathbf{P}_{N+1,i}\} = \left\{\begin{bmatrix} \mathbf{T}_{N,i} \\ \mathbf{t}_i^{(*)} \end{bmatrix}; \begin{bmatrix} \mathbf{P}_{N,i} \\ p_i^{(*)} \end{bmatrix}\right\}, \quad i = 1, ..., I.$$

(d) Set $N = N + 1$ and go to (2a); Stop when $N = N_{max}$.

3. **Support Vector Regression (SVR).**

(a) Generate M (previously-unseen) test configurations of the crack

$$\mathbf{P}_M = \begin{bmatrix} \mathbf{p}^{(1)} \\ \vdots \\ \mathbf{p}^{(M)} \end{bmatrix} = \begin{bmatrix} p_1^{(1)} & ... & p_I^{(1)} \\ \vdots & \ddots & \vdots \\ p_1^{(M)} & ... & p_I^{(M)} \end{bmatrix} = \begin{bmatrix} x_0^{(1)} & y_0^{(1)} & z_0^{(1)} \\ \vdots & \vdots & \vdots \\ x_0^{(M)} & y_0^{(M)} & z_0^{(M)} \end{bmatrix}$$

EEDIA Research Center

(b) Use the forward solver to compute the ECT signal associated to the M test samples. Build the following matrix of measurements (dim. $[M \times 2K] = [M \times F]$)

$$\mathbf{\Psi}_M = \begin{bmatrix} \mathbf{\Psi}^{(1)} \\ \vdots \\ \mathbf{\Psi}^{(M)} \end{bmatrix} = \begin{bmatrix} \Re\left\{\Psi_1\left(\mathbf{p}^{(1)}\right)\right\} & \Im\left\{\Psi_1\left(\mathbf{p}^{(1)}\right)\right\} & ... & \Re\left\{\Psi_K\left(\mathbf{p}^{(1)}\right)\right\} & \Im\left\{\Psi_K\left(\mathbf{p}^{(1)}\right)\right\} \\ \vdots & & \ddots & & \vdots \\ \Re\left\{\Psi_1\left(\mathbf{p}^{(M)}\right)\right\} & \Im\left\{\Psi_1\left(\mathbf{p}^{(M)}\right)\right\} & ... & \Re\left\{\Psi_K\left(\mathbf{p}^{(M)}\right)\right\} & \Im\left\{\Psi_K\left(\mathbf{p}^{(M)}\right)\right\} \end{bmatrix}$$

(c) For each i-th parameter of the crack:

i. Use matrix $\mathbf{W}_i$to project $\mathbf{\Psi}_M$ onto the reduced J-dimensional space

$$\mathbf{T}_{M,i} = \left(\mathbf{\Psi}_M - \overline{\mathbf{\Psi}_{N_1}}\right) \times \mathbf{W}_i = \begin{bmatrix} \mathbf{t}_i^{(1)} \\ \vdots \\ \mathbf{t}_i^{(M)} \end{bmatrix} = \begin{bmatrix} t_{i,1}^{(1)} & ... & t_{i,J}^{(1)} \\ \vdots & \ddots & \vdots \\ t_{i,1}^{(M)} & ... & t_{i,J}^{(M)} \end{bmatrix}, \quad i = 1, ..., I$$

ii. Train a SVR using the generated training set of N training input-output pairs composed as

$$\{\mathbf{T}_N; \mathbf{P}_{N,i}\} = \left\{ \begin{bmatrix} t_1^{(1)} & ... & t_J^{(1)} \\ \vdots & \ddots & \vdots \\ t_1^{(N)} & ... & t_J^{(N)} \end{bmatrix}; \begin{bmatrix} p_i^{(1)} \\ \vdots \\ p_i^{(N)} \end{bmatrix} \right\}; \quad i = 1, ..., I$$

iii. Test the SVR giving it as input the transformed features associated to the M test samples $\mathbf{T}_{M,i}$. As output, the SVR will produce a vector of M estimated values for the i-th parameter

$$\widetilde{\mathbf{P}}_{M,i} = \begin{bmatrix} \widetilde{p}_i^{(1)} \\ \vdots \\ \widetilde{p}_i^{(M)} \end{bmatrix}.$$

1.2 Critical aspects of algorithm #1

1. **Problem 1: evaluation of the C candidates with the forward solver during the generation of the training sets.**

- In order to add a new training sample to the I training sets, the forward solver is used to compute the ECT signal associated to each candidate sample ($c = 1, ..., C$) in order to determine the projected features in the reduced space and select the best candidate.
- This is a huge waste of computational resources if the forward solver is a full-wave em solver (e.g., $CIVA$);
- In order to generate N training samples for a given parameter of the crack, we need to call the forward solver $N \times C$ times!;
- This problem is much less significant if the forward solver is not a full-wave solver but a forward meta-model (much faster, but needs its own training phase...);

ELEDIA Research Center

2. **Problem 2: treating each parameter to estimate separately during the generation of the training sets.**

   - We apply the PLS algorithm for each parameter to estimate ($i = 1, ..., I$);
   - This leads to the generation of I different reduced spaces, that are filled using the OSF strategy independently;
   - The training samples generated for each parameter are thus different (one candidate sample can be good for one parameter but not for the others...).

Given the above considerations, two evolutions of this original algorithm are presented in the following, trying to solve the two critical aspects.

1.3 Algorithm #2: Solution to problem 1

Goal: avoid to use the forward solver to compute the ECT signal associated to the C candidate samples during the generation of the training sets.

Solution: instead of calling the forward solver to compute the ECT signal associated to the C candidates, the following steps are performed

1. For each parameter to estimate ($i = 1, ..., I$) use a linear interpolator (fast!) to predict the matrix of J-dimensional features associated to the C candidates

$$\widetilde{\mathbf{T}}_{C,i} = \begin{bmatrix} \widetilde{\mathbf{t}}_i^{(1)} \\ \vdots \\ \widetilde{\mathbf{t}}_i^{(C)} \end{bmatrix} = \begin{bmatrix} \widetilde{t}_{i,1}^{(1)} & \dots & \widetilde{t}_{i,J}^{(1)} \\ \vdots & \ddots & \vdots \\ \widetilde{t}_{i,1}^{(C)} & \dots & \widetilde{t}_{i,J}^{(C)} \end{bmatrix}, \quad i = 1, ..., I$$

   the interpolation uses the information from the previously collected samples;

2. Select the candidate that maximizes the minimum distance w.r.t. the previously collected samples in the reduced space of features. Use the predicted values at the previous step.

3. Once the best candidate $\widetilde{\mathbf{t}}_i^{(*)}$ has been selected, compute the real $\mathbf{t}_i^{(*)}$ by calling the forward solver and projecting the vector of measurements onto the reduced space and append it to the previous training set.

Observations

- By using a linear interpolator for estimating the J-dimensional features associated to the C candidates allows to call the forward solver only for computing the measurements associated to the selected candidate.
- In order to generate N training samples for a given parameter of the crack, we need to call the forward solver only N times;

EDF R&D Research Center

1.4 Algorithm #3: Solution to problems 1 and 2

Goal: avoid to treat each parameter to estimate separately. Generate a unique set of features that are "good" training samples for all the parameters.

Solution: apply the PLS algorithm jointly considering the I parameters of the crack.

1. **Partial Least Squares (PLS)**

   (a) Build an initial set $\mathbf{P}_{N_1}$ of N_1 configurations of the crack (dim. $[N_1 \times I]$) (e.g., positions within the plate)

   $$\mathbf{P}_{N_1} = \begin{bmatrix} \mathbf{p}^{(1)} \\ \vdots \\ \mathbf{p}^{(N_1)} \end{bmatrix} = \begin{bmatrix} p_1^{(1)} & \dots & p_I^{(1)} \\ \vdots & \ddots & \vdots \\ p_1^{(N_1)} & \dots & p_I^{(N_1)} \end{bmatrix} = \begin{bmatrix} x_0^{(1)} & y_0^{(1)} & z_0^{(1)} \\ \vdots & \vdots & \vdots \\ x_0^{(N_1)} & y_0^{(N_1)} & z_0^{(N_1)} \end{bmatrix}$$

   using a uniform grid sampling in the I-dimensional space of crack parameters;

   (b) Use the forward solver $\Phi\{.\}$ to compute the ECT signal in K measurement points associated to the N_1 configurations of the crack. Build the following matrix of measurements where real and imaginary parts of each measurement point are treated as separate real-valued features (dim. $[N_1 \times 2K] = [N_1 \times F]$)

   $$\mathbf{\Psi}_{N_1} = \begin{bmatrix} \mathbf{\Psi}^{(1)} \\ \vdots \\ \mathbf{\Psi}^{(N_1)} \end{bmatrix} = \begin{bmatrix} \Re\{\Psi_1(\mathbf{p}^{(1)})\} & \Im\{\Psi_1(\mathbf{p}^{(1)})\} & \dots & \Re\{\Psi_K(\mathbf{p}^{(1)})\} & \Im\{\Psi_K(\mathbf{p}^{(1)})\} \\ \vdots & & \ddots & & \vdots \\ \Re\{\Psi_1(\mathbf{p}^{(N_1)})\} & \Im\{\Psi_1(\mathbf{p}^{(N_1)})\} & \dots & \Re\{\Psi_K(\mathbf{p}^{(N_1)})\} & \Im\{\Psi_K(\mathbf{p}^{(N_1)})\} \end{bmatrix};$$

   (c) Use PLS to extract from matrix $\mathbf{\Psi}_{N_1}$ $J << F$ features to be used during the inversion. The PLS algorithm receives as input the pair of matrices

   $$\{\mathbf{\Psi}_{N_1}; \mathbf{P}_{N_1}\}$$

   and retrieves a matrix $\mathbf{T}_{N_1}$ of extracted J-dimensional features (dim. $[N_1 \times J]$) that are the most informative ones for predicting $\mathbf{P}_{N_1}$

   $$\mathbf{T}_{N_1} = \begin{bmatrix} \mathbf{t}^{(1)} \\ \vdots \\ \mathbf{t}^{(N_1)} \end{bmatrix} = \begin{bmatrix} t_1^{(1)} & \dots & t_J^{(1)} \\ \vdots & \ddots & \vdots \\ t_1^{(N_1)} & \dots & t_J^{(N_1)} \end{bmatrix}$$

   as well as the matrix $\mathbf{W}$ of weights (dim. $[F \times J]$) used to perform a linear projection of $\mathbf{\Psi}_{N_1}$ into $\mathbf{T}_{N_1}$

   $$\mathbf{T}_{N_1} = \left(\mathbf{\Psi}_{N_1} - \overline{\mathbf{\Psi}_{N_1}}\right) \times \mathbf{W}$$

ELEDIA Research Center

where $\overline{\mathbf{\Psi}_{N_1}}$ is a $[N_1 \times F]$ matrix with rows containing the average values of the columns of $\mathbf{\Psi}_{N_1}$

$$\overline{\mathbf{\Psi}_{N_1}} = \begin{bmatrix} \overline{\Re\{\Psi_1\}} & \overline{\Im\{\Psi_1\}} & ... & \overline{\Re\{\Psi_K\}} & \overline{\Im\{\Psi_K\}} \\ \vdots & & \ddots & & \vdots \\ \overline{\Re\{\Psi_1\}} & \overline{\Im\{\Psi_1\}} & ... & \overline{\Re\{\Psi_K\}} & \overline{\Im\{\Psi_K\}} \end{bmatrix};$$

and

$$\begin{cases} \overline{\Re\{\Psi_k\}} = \frac{1}{N_1}\sum_{n=1}^{N_1} \Re\left\{\Psi_k\left(\mathbf{p}^{(n)}\right)\right\} \\ \overline{\Im\{\Psi_k\}} = \frac{1}{N_1}\sum_{n=1}^{N_1} \Im\left\{\Psi_k\left(\mathbf{p}^{(n)}\right)\right\} \end{cases} k = 1, ..., K$$

(d) The initial training set of N_1 samples is then composed by the input-output pair

$$\{\mathbf{T}_{N_1}; \mathbf{P}_{N_1}\} = \left\{ \begin{bmatrix} t_1^{(1)} & ... & t_J^{(1)} \\ \vdots & \ddots & \vdots \\ t_1^{(N_1)} & ... & t_J^{(N_1)} \end{bmatrix}; \begin{bmatrix} p_1^{(1)} & ... & p_I^{(1)} \\ \vdots & \ddots & \vdots \\ p_1^{(N_1)} & ... & p_I^{(N_1)} \end{bmatrix} \right\}.$$

2. **Output Space Filling (OSF).** The goal is to select new samples in the input space of crack parameters such that the reduced space of extracted features is uniformly sampled. Set $N = N_1$. To generate a new training sample perform the following steps

(a) Generate a set of C "candidate" configurations of the crack, by sampling with LHS the I-dimensional space of crack parameters

$$\mathbf{P}_C = \begin{bmatrix} \mathbf{p}^{(1)} \\ \vdots \\ \mathbf{p}^{(C)} \end{bmatrix} = \begin{bmatrix} p_1^{(1)} & ... & p_I^{(1)} \\ \vdots & \ddots & \vdots \\ p_1^{(C)} & ... & p_I^{(C)} \end{bmatrix} = \begin{bmatrix} x_0^{(1)} & y_0^{(1)} & z_0^{(1)} \\ \vdots & \vdots & \vdots \\ x_0^{(C)} & y_0^{(C)} & z_0^{(C)} \end{bmatrix}$$

(b) Apply a linear interpolator on the previously-collected pair $\{\mathbf{P}_N; \mathbf{T}_N\}$ to estimate the matrix of J-dimensional features $\mathbf{T}_C$ associated to $\mathbf{P}_C$

$$\widetilde{\mathbf{T}}_C = \begin{bmatrix} \widetilde{\mathbf{t}}^{(1)} \\ \vdots \\ \widetilde{\mathbf{t}}^{(C)} \end{bmatrix} = \begin{bmatrix} \widetilde{t}_1^{(1)} & ... & \widetilde{t}_J^{(1)} \\ \vdots & \ddots & \vdots \\ \widetilde{t}_1^{(C)} & ... & \widetilde{t}_J^{(C)} \end{bmatrix}$$

we use linear interpolation because we cannot call the forward operator to compute the ECT measurements associated to each candidate sample (too expensive!);

(c) Compute the Euclidean distance between each c-th estimated vector of extracted features $\widetilde{\mathbf{t}}^{(c)} (c = 1, ..., C)$ and the vectors $\mathbf{t}^{(n)}, n = 1, ..., N$ previously collected (i.e., the rows of $\mathbf{T}_N$)

$$dist\left\{\widetilde{\mathbf{t}}^{(c)}, \mathbf{t}^{(n)}\right\} = \sqrt{\sum_{j=1}^{J}\left(\widetilde{t}_j^{(c)} - t_j^{(n)}\right)^2}, \quad c = 1, ..., C, \quad n = 1, ..., N$$

EEDIA Research Center

(d) Select the vector of estimated extracted features $\widetilde{\mathbf{t}}^{(*)}$ having the maximum minimum distance with respect to the previously collected N vectors $(\mathbf{t}^{(n)}, n = 1, ..., N)$.

$$\widetilde{\mathbf{t}}^{(*)} = arg\left\{\max_{c=1,...,C}\left[\min_{n=1,...,N}\left(dist\left\{\widetilde{\mathbf{t}}^{(c)}, \mathbf{t}^{(n)}\right\}\right)\right]\right\}$$

This vector corresponds to the candidate sample $\mathbf{p}^{(*)}$ that is the farthest one in the space of extracted features.

(e) Use the forward solver to compute the ECT signal associated to the selected candidate sample $\mathbf{p}^{(*)}$. Then build the following vector of measurements (dim. $[1 \times 2K] = [1 \times F]$)

$$\mathbf{\Psi}^{(*)} = \left[\begin{array}{cccc} \Re\left\{\Psi_1\left(\mathbf{p}^{(*)}\right)\right\} & \Im\left\{\Psi_1\left(\mathbf{p}^{(*)}\right)\right\} & ... & \Re\left\{\Psi_K\left(\mathbf{p}^{(*)}\right)\right\} \quad \Im\left\{\Psi_K\left(\mathbf{p}^{(*)}\right)\right\} \end{array}\right]$$

(f) Use the transformation matrix $\mathbf{W}$ computed at the initial step to project $\mathbf{\Psi}^{(*)}$ onto the reduced J-dimensional space of extracted features

$$\mathbf{t}^{(*)} = \left(\mathbf{\Psi}^{(*)} - \overline{\mathbf{\Psi}_{N_1}}\right) \times \mathbf{W} = \left[\begin{array}{ccc} t_1^{(*)} & ... & t_J^{(*)} \end{array}\right]$$

(g) Append the new vector of extracted features $\mathbf{t}^{(*)}$ and the associated vector of crack parameters $\mathbf{p}^{(*)}$ to the training set, that will be composed as

$$\{\mathbf{T}_{N+1}; \mathbf{P}_{N+1}\} = \left\{\left[\begin{array}{c} \mathbf{T}_{N_1} \\ \mathbf{t}^{(*)} \end{array}\right]; \left[\begin{array}{c} \mathbf{P}_{N_1} \\ \mathbf{p}^{(*)} \end{array}\right]\right\}.$$

(h) Set $N = N + 1$ and go to (2a). Stop when $N = N_{max}$.

3. **Support Vector Regression (SVR).**

(a) Generate M (previously-unseen) test configurations of the crack

$$\mathbf{P}_M = \left[\begin{array}{c} \mathbf{p}^{(1)} \\ \vdots \\ \mathbf{p}^{(M)} \end{array}\right] = \left[\begin{array}{ccc} p_1^{(1)} & ... & p_I^{(1)} \\ \vdots & \ddots & \vdots \\ p_1^{(M)} & ... & p_I^{(M)} \end{array}\right] = \left[\begin{array}{ccc} x_0^{(1)} & y_0^{(1)} & z_0^{(1)} \\ \vdots & \vdots & \vdots \\ x_0^{(M)} & y_0^{(M)} & z_0^{(M)} \end{array}\right]$$

(b) Use the forward solver to compute the ECT signal associated to the M test samples. Build the following matrix of measurements (dim. $[M \times 2K] = [M \times F]$)

$$\mathbf{\Psi}_M = \left[\begin{array}{c} \mathbf{\Psi}^{(1)} \\ \vdots \\ \mathbf{\Psi}^{(M)} \end{array}\right] = \left[\begin{array}{cccc} \Re\left\{\Psi_1\left(\mathbf{p}^{(1)}\right)\right\} & \Im\left\{\Psi_1\left(\mathbf{p}^{(1)}\right)\right\} & ... & \Re\left\{\Psi_K\left(\mathbf{p}^{(1)}\right)\right\} \quad \Im\left\{\Psi_K\left(\mathbf{p}^{(1)}\right)\right\} \\ \vdots & & \ddots & \vdots \\ \Re\left\{\Psi_1\left(\mathbf{p}^{(M)}\right)\right\} & \Im\left\{\Psi_1\left(\mathbf{p}^{(M)}\right)\right\} & ... & \Re\left\{\Psi_K\left(\mathbf{p}^{(M)}\right)\right\} \quad \Im\left\{\Psi_K\left(\mathbf{p}^{(M)}\right)\right\} \end{array}\right]$$

ELEDIA Research Center

(c) Use the transformation matrix $\mathbf{W}$ to project the matrix of measurements $\mathbf{\Psi}_M$ onto the reduced J-dimensional space

$$\mathbf{T}_M = \left(\mathbf{\Psi}_M - \overline{\mathbf{\Psi}_{N_1}}\right) \times \mathbf{W} = \begin{bmatrix} \mathbf{t}^{(1)} \\ \vdots \\ \mathbf{t}^{(M)} \end{bmatrix} = \begin{bmatrix} t_1^{(1)} & ... & t_J^{(1)} \\ \vdots & \ddots & \vdots \\ t_1^{(M)} & ... & t_J^{(M)} \end{bmatrix}$$

(d) For a given training dimension N and for each i-th parameter of the crack to estimate:

i. Train a SVR using a training set composed as

$$\{\mathbf{T}_N; \mathbf{P}_{N,i}\} = \left\{ \begin{bmatrix} t_1^{(1)} & ... & t_J^{(1)} \\ \vdots & \ddots & \vdots \\ t_1^{(N)} & ... & t_J^{(N)} \end{bmatrix}; \begin{bmatrix} p_i^{(1)} \\ \vdots \\ p_i^{(N)} \end{bmatrix} \right\}; \quad i = 1, ..., I$$

where $\mathbf{P}_{N,i}$ is the i-th column of matrix $\mathbf{P}_N$;

ii. Test the SVR giving it as input the transformed features associated to the M test samples $\mathbf{T}_M$. As output, the SVR will produce a vector of M estimated values for the i-th parameter

$$\widetilde{\mathbf{P}}_{M,i} = \begin{bmatrix} \widetilde{p}_i^{(1)} \\ \vdots \\ \widetilde{p}_i^{(M)} \end{bmatrix}.$$

1.4.1 Parameters of interest

1. The dimension of the initial training set: N_1;
2. The number of extracted PLS components: J;
3. The number of measurement points: K;
4. The max dimension of the training set: N_{max};

ELEDIA Research Center

2 Problem 1: Crack Location Estimation Inside a Plate Structure

2.1 Parameters of the forward solver (fixed)

- **Forward solver**
  - type: $OSF - Kriging$ metamodel provided by $CEA - LIST$;
  - total number of measurement points along x (i.e., across the crack): $H_x = 41$;
  - measurement step along x: $\Delta_x = 0.5$ [mm];
  - total extension of the measurement region along x: $L_x = 20.0$ [mm];
  - total number of measurement points along y (i.e., along the crack): $H_y = 57$;
  - measurement step along y: $\Delta_y = 0.5$ [mm];
  - total extension of the measurement region along y: $L_y = 28.0$ [mm];
  - total number of measurement point computed by the forward solver: $H = H_x \times H_y = 2337$;

Plate	
Thickness T	1.55 [mm]
Conductivity σ	1.02 [MS/m]
Coil	
Inner radius r_1	1.0 [mm]
Outer radius r_2	1.75 [mm]
Length l_c	2.0 [mm]
Number of turns n_t	328
Lift-off δ	0.303 [mm]
Frequency f	100.0 [KHz]
Crack	
Depth d_0	0.62 [mm] (40% of T)
Length l_0	10.0 [mm]
Width w_0	0.3 [mm]

Table 1: Fixed parameters.

Parameter	**Min [mm]**	**Max [mm]**
Crack x-coordinate x_0	5.0	25.0
Crack y-coordinate y_0	1.0	29.0
Crack z-coordinate z_0	0.93	1.24

Table 2: Validity ranges of the forward meta-model.

ELEDIA Research Center

2.2 $PLS-OSF-SVR$ (Algorithm #3): Analysis for $J=10$ - Performances

2.2.1 Parameters

- **Measurement set-up for the inversion**
  - considered measurement step: $\Delta_x = \Delta_y = 1.0$ [mm];
  - number of considered measurement points $K = K_x \times K_y = 21 \times 29 = 609$;
  - measured quantity for each k-th point: $\{\Re(\Psi_k), \Im(\Psi_k)\}$;
  - total number of measured features: $F = 2 \times K = 1218$;

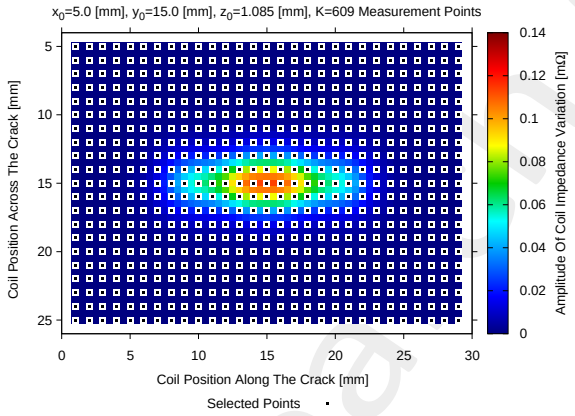

Figure 1: Location of the measurement points selected for the inversion ($K = 609$).

- $PLS-OSF-SVR$ **(algorithm #3)**
  - Initial training set (uniform grid)
    - Number of quantization levels: $Q_{x_0} = Q_{y_0} = Q_{z_0} = 5$;
    - Number of initial training samples: $N_1 = Q_{x_0} \times Q_{y_0} \times Q_{z_0} = 125$;
  - Number of extracted PLS components: $J = 10$;
  - SNR on training measurements: noiseless data;
  - Number of candidate samples: $C = 150$ $(50 \times I)$ (generated via LHS sampling);
  - SNR on training data: Noiseless;
  - Test set generation
    - Sampling: Latin Hypercube Sampling (LHS);
    - Number of test samples: $M = 1000$;
    - SNR on test data: Noiseless + $SNR = \{40; 30; 20; 10\}$ [dB].

ELEDIA Research Center

2.2.2 Calibration of the SVR parameters via cross-validation

The best (C, γ) pair of parameters is selected for training the three SVR regressors.

Parameters

- number of subsets: $V = 5$;
- variation range for parameter C: $C \in \left\{10^0; 10^1; ...; 10^8\right\}$;
- variation range for parameter γ: $\gamma \in \left\{10^{-6}; 10^{-5}; ...; 10^1\right\}$;
- dimension of the training set: $N = 1000$;

Results

Parameter	**Best C ($C*$)**	**Best γ ($\gamma*$)**	**CV MSE (η)**
Crack x-coordinate x_0	10^5	10^{-1}	1.31×10^{-2}
Crack y-coordinate y_0	10^2	1	2.65×10^{-2}
Crack z-coordinate z_0	10^5	10^{-2}	4.25×10^{-3}

Table 3: Optimal (C, γ) pairs and $CV\ MSE$ found by applying a 5-fold cross-validation for the estimation of the crack coordinates.

ELEDIA Research Center

2.2.3 Estimation of x_0: PLS-OSF-SVR vs. Standard Approach

$SNR = 20$ **[dB] on** ECT **Measurements**

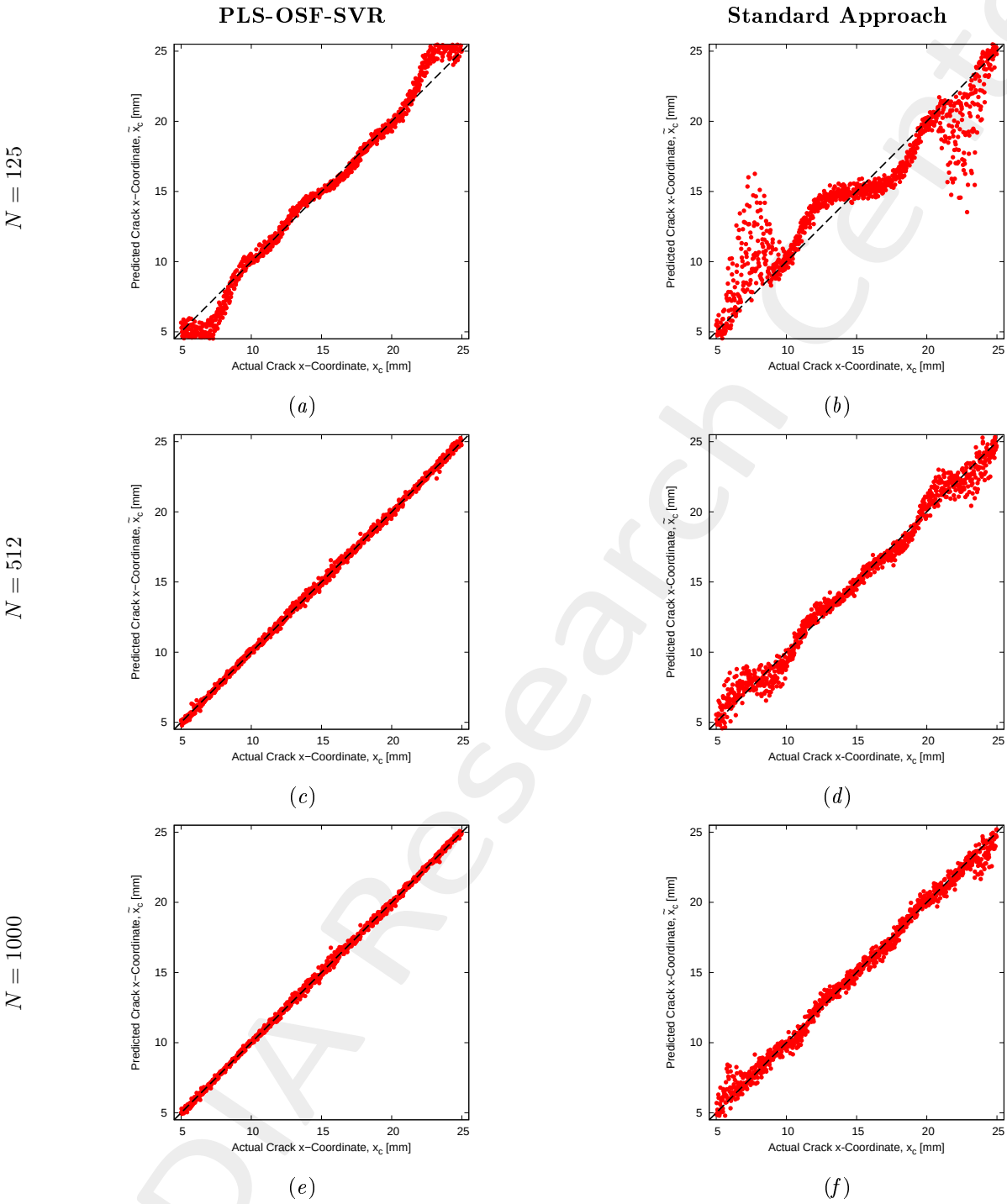

Figure 2: **PLS-OSF-SVR vs Standard Approach** - Actual vs. predicted x-coordinate of the crack for different values of N. $SNR = 20$ [dB] on test ECT data.

ELEDIA Research Center

2.2.4 Estimation of y_0: PLS-OSF-SVR vs. Standard Approach

$SNR = 20$ **[dB] on** ECT **Measurements**

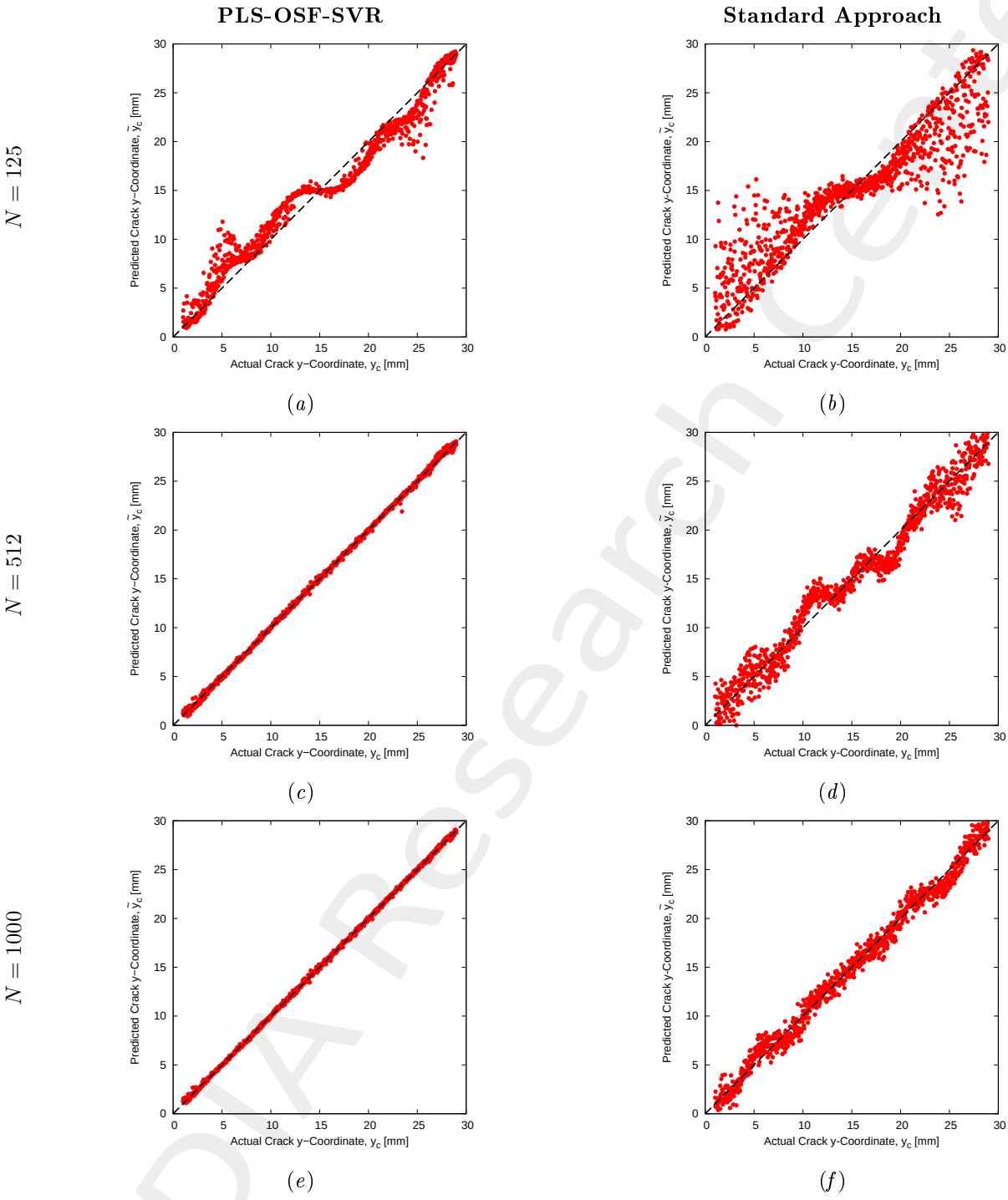

Figure 3: **PLS-OSF-SVR vs Standard Approach** - Actual vs. predicted y-coordinate of the crack for different values of N. $SNR = 20$ [dB] test ECT data.

ELEDIA Research Center

2.2.5 Estimation of z_0: PLS-OSF-SVR vs. Standard Approach

$SNR = 20$ **[dB] on** ECT **Measurements**

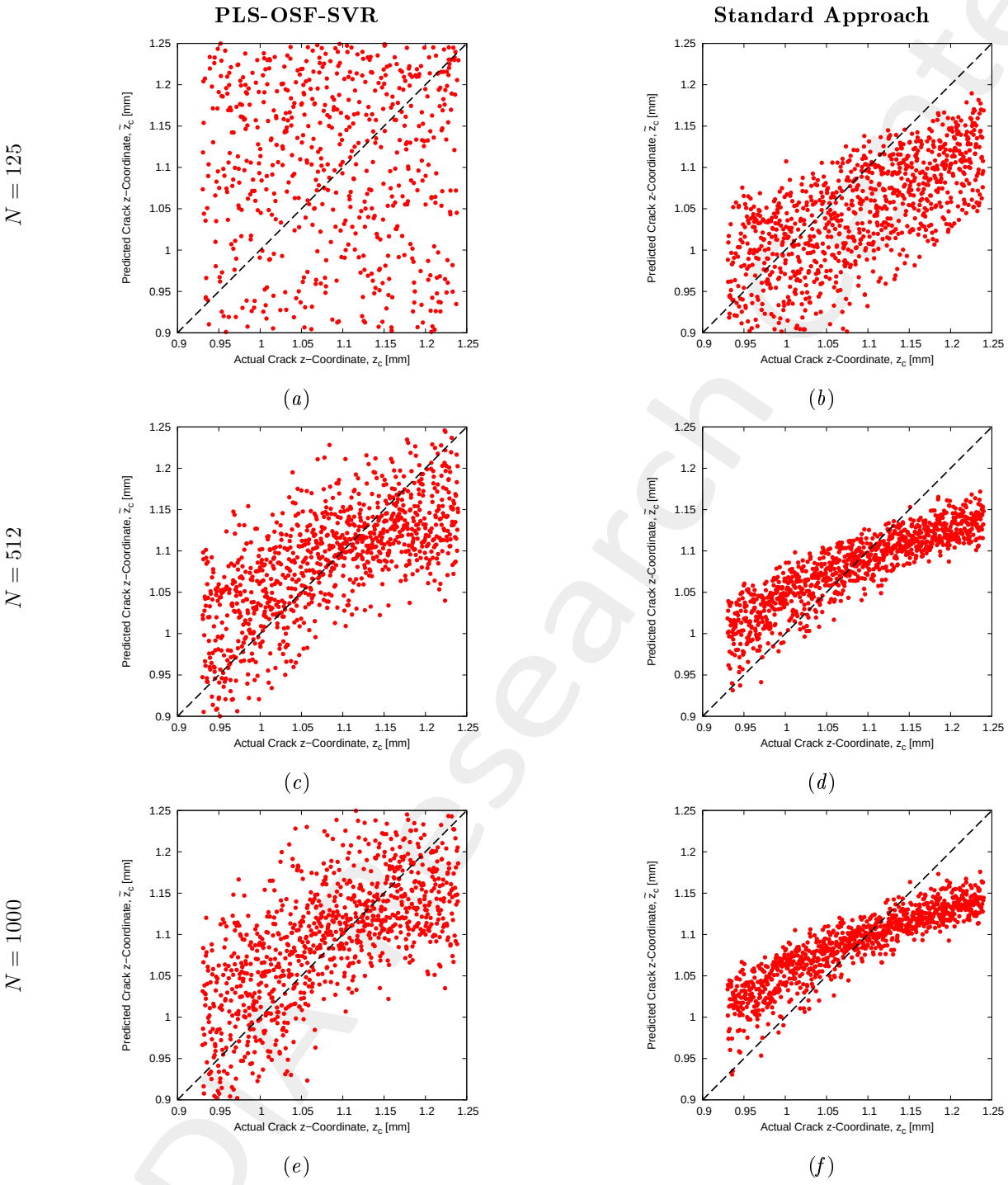

Figure 4: **PLS-OSF-SVR vs Standard Approach** - Actual vs. predicted z-coordinate of the crack for different values of N. $SNR = 20$ [dB] on test ECT data.

ELDA Research Center

2.2.6 Errors

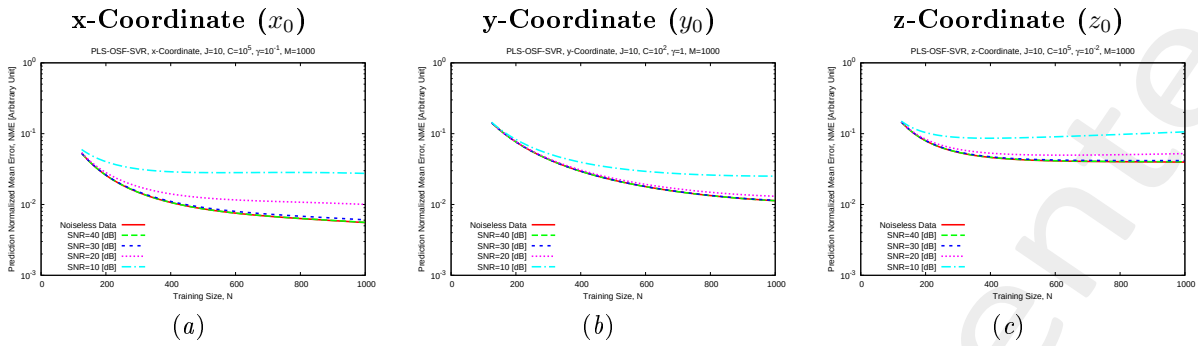

Figure 5: **PLS-OSF-SVR** - Normalized Mean Error (NME) vs. training size (N)

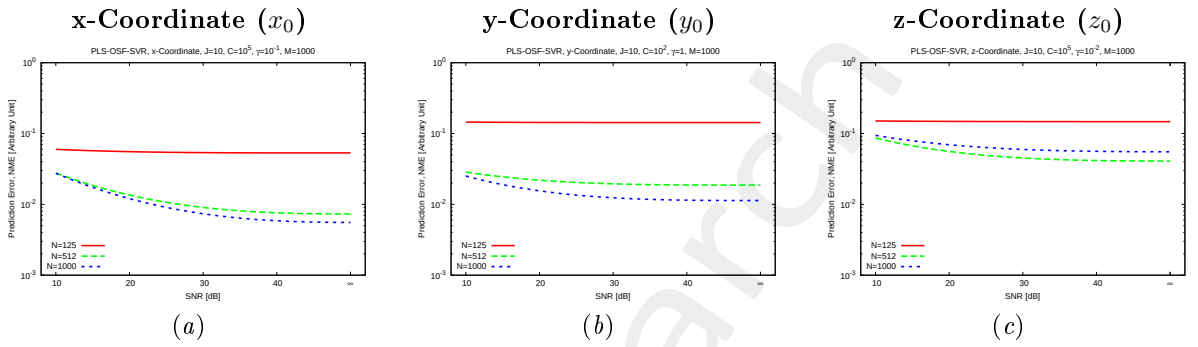

Figure 6: **PLS-OSF-SVR** - Normalized Mean Error (NME) vs. SNR on the test ECT measurements.

2.2.7 Conclusions

The results in this section show that

- the performances of the $PLS-OSF-SVR$ are significantly higher, in terms of prediction error, w.r.t. the standard approach, for a given dimension N of the training set and SNR on test data.
- This is particularly evident for the estimation of the x and y coordinates;
- the prediction performances are not only enhanced thanks to the compression of the ECT measurements in a much lower number of informative features, but also by the ability of the OSF technique to uniformly sample the reduced space of extracted features.

ELEDIA Research Center

References

[1] M. Salucci, N. Anselmi, G. Oliveri, P. Calmon, R. Miorelli, C. Reboud, and A. Massa, "Real-time NDT-NDE through an innovative adaptive partial least squares SVR inversion approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 11, pp. 6818-6832, Nov. 2016.

[2] M. Salucci, G. Oliveri, F. Viani, R. Miorelli, C. Reboud, P. Calmon, and A. Massa, "A learning-by-examples approach for non-destructive localization and characterization of defects through eddy current testing measurements," in *2015 IEEE International Symposium on Antennas and Propagation*, Vancouver, 2015, pp. 900-901.

[3] M. Salucci, S. Ahmed and A. Massa, "An adaptive Learning-by-Examples strategy for efficient Eddy Current Testing of conductive structures," in *2016 European Conference on Antennas and Propagation*, Davos, 2016, pp. 1-4.

[4] P. Rocca, M. Benedetti, M. Donelli, D. Franceschini, and A. Massa, "Evolutionary optimization as applied to inverse problems," *Inverse Probl.*, vol. 25, pp. 1-41, Dec. 2009.

[5] A. Massa, P. Rocca, and G. Oliveri, "Compressive sensing in electromagnetics - A review," *IEEE Antennas Propag. Mag.*, pp. 224-238, vol. 57, no. 1, Feb. 2015.

[6] N. Anselmi, G. Oliveri, M. Salucci, and A. Massa, "Wavelet-based compressive imaging of sparse targets," *IEEE Trans. Antennas Propag.*, vol. 63, no. 11, pp. 4889-4900, Nov. 2015.

[7] M. Salucci, G. Oliveri, and A. Massa, "GPR prospecting through an inverse-scattering frequency-hopping multifocusing approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 12, pp. 6573-6592, Dec. 2015.

[8] T. Moriyama, G. Oliveri, M. Salucci, and T. Takenaka, "A multi-scaling forward-backward time-stepping method for microwave imaging," *IEICE Electron. Express*, vol. 11, no. 16, pp. 1-12, Aug. 2014.

[9] T. Moriyama, M. Salucci, M. Tanaka, and T. Takenaka, "Image reconstruction from total electric field data with no information on the incident field," *J. Electromagnet. Wave.*, vol. 30, no. 9, pp. 1162-1170, 2016.

[10] M. Salucci, L. Poli, and A. Massa, "Advanced multi-frequency GPR data processing for non-linear deterministic imaging," *Signal Processing - Special Issue on "Advanced Ground-Penetrating Radar Signal-Processing Techniques,"* vol. 132, pp. 306-318, Mar. 2017.

ELEDIA Research Center