SVM-based Multi-point Prediction System of Solar Irradiance using DTW-based Cluster Analysis*

Yukiya Tanaka†§ and Masaki Takahashi‡§

Unexpected short-term fluctuation of solar irradiance may negatively affect an electric power system; hence, predicting the solar irradiance is necessary. Considering that large quantities of solar power will be introduced across wide areas in the future, this paper addresses multi-point predictions of solar irradiance. Support vector machines (SVMs) perform well for predicting solar irradiance, but involve relatively high computational complexity. In addition, SVMs have to be repeatedly computed at multiple points because prediction models need to be updated regularly to adapt to the season. Here, we introduce a multi-point prediction system that considerably reduces the amount of calculation. Instead of constructing prediction models at all grid points, this system establishes clusters based on similarities of time series data by executing the dynamic time warping algorithm, and then constructs a small number of models representing each cluster. Further, it automatically tunes the number of clusters by inspecting prediction accuracy. Simulation results reveal that the system saves a considerable amount of calculation while maintaining high prediction accuracy.

Nomenclature

- \( \alpha_p \): Multi-point time series of solar irradiance data \( \text{W/m}^2 \)
- \( p, p' \): Grid point indices
- \( L_p \): Length of time series data
- \( N \): Number of clusters
- \( D_{p,p'} \): Accumulated cost matrix \( \text{W/m}^2 \)
- \( l, l' \): Row and column indices
- \( T \): Sakoe-Chiba band width
- \( d_{\text{DTW}}(\cdot,\cdot) \): DTW distance between two time series data \( \text{W/m}^2 \)
- \( C, C' \): Clusters
- \( d_{\text{UPGMA}}(\cdot,\cdot) \): Distance between two clusters using UPGMA
- \( x_{p,m} \): Learning input data \( \text{W/m}^2 \)
- \( y_{p,m} \): Learning output data \( \text{W/m}^2 \)
- \( m, m' \): Learning data index
- \( n_l \): Number of learning data
- \( w_p \): Weight vector
- \( b_p \): Bias term \( \text{W/m}^2 \)
- \( f_p(\cdot) \): Autoregressive model
- \( x \): General input vector \( \text{W/m}^2 \)
- \( C_p \): Regularization parameter
- \( \varepsilon_p \): Abandoned error value \( \text{W/m}^2 \)
- \( \xi_{p,m}, \xi_{p,m}' \): Slack variables \( \text{W/m}^2 \)
- \( \alpha_{p,m}, \alpha_{p,m}' \): Lagrange multipliers
- \( x_{p,t} \): Test input data \( \text{W/m}^2 \)
- \( y_{p,t} \): Test output data \( \text{W/m}^2 \)
- \( t \): Test data index
- \( n_t \): Number of test data
- \( q \): Inspection parameter \( \text{W/m}^2 \)
- \( N_f \): Tuned number of clusters

1. Introduction

Recent global environmental and energy concerns have revealed the need for renewable energy such as solar power. However, solar irradiance fluctuates depending on the season, time, and climate. This fluctuation needs to be suppressed to stabilize the electric power supply. Operational planning usually involves controlling generators; however, unexpected fluctua-
tion of the solar power may negatively affect an existing power system. Therefore, it is necessary to predict the solar irradiance. The prediction of short-term solar irradiance has been actively studied to enhance planning effectiveness. Short-term prediction using a support vector machine (SVM) has been proposed [2,3]. Others investigated implementing forecasting at a real photovoltaic power plant [4]. Recently, hybrid models based on some algorithms have been proposed [5,6]. A comparison of different methods [7], very short-term prediction [8,9], and online prediction [10,11] have also been researched. Currently, most research has considered single-point predictions of solar power; however, it is expected that large quantities of solar power will be introduced across wide areas in the future. Therefore, this paper addresses multi-point predictions of solar irradiance over a wide area.

SVMs represent one of the leading methods for predicting solar irradiance [2–5,12–14]. SVMs use historical solar irradiance data to construct prediction models. On the other hand, earth observation satellites can provide solar irradiance data at multiple points [15–17]. Further, recent development of AMATERASS [18], a solar radiation product based on observational satellite data, has made it possible to observe multi-point solar irradiance across all of Japan.

Technologies available for observing the earth continue to improve. Himawari-8 is a new weather satellite that has 0.5-km spatial resolution and 10-min temporal resolution over Japan, two and three times greater, respectively, than Himawari-7 [19]. Thus, it is an ideal data source for construction of prediction models for grid points. However, computational overheads are high, because SVMs involve relatively high computational complexity. Further, SVMs have to be repeatedly computed at multiple points because prediction models need to be updated regularly to adapt to the season. Because developing countries do not possess many computational resources, considerably reducing the amount of calculation is necessary.

This paper introduces a multi-point prediction system that reduces the amount of calculation. The use of an inefficient method to construct prediction models at all grid points is highly time consuming. To improve efficiency, instead of constructing models at all grid points, this system uses time series analysis with less computational complexity and establishes clusters for constructing models. Specifically, this system initially measures the similarities between two time series of solar irradiance data by executing the dynamic time warping (DTW) algorithm. Then, it performs clustering for all grid points based on the similarities, before constructing prediction models to represent each cluster. In short, this system establishes clusters based on similarities of time series data in order to reduce the number of models. The system measures the similarities in an attempt to maintain the prediction accuracy as high as possible.

We have previously presented the basic scheme of a multi-point prediction system [1]. However, we encountered some problems in practically realizing this system. These problems relate to: reduce further the amount of calculation, managing an increase in the number of grid points, and automating an algorithm. In this paper, we solve these problems and aim to realize a practical multi-point prediction system.

2. Multi-point Prediction System

2.1 Outline

Flow of multi-point prediction system is shown in Fig. 1. Input into the system comprises historical multi-point time series solar irradiance data \( a_p = (a_{p,1}, \ldots, a_{p,L_p}) \), where \( p \) is the index of a grid point, \( L_p \) is the length of \( p \)-th time series data. The output comprises the predicted values at multiple points. The number of clusters \( N \) is automatically determined.

(1) Measuring similarities of time series data

The system first measures similarities between two past time series of solar irradiance data by running the DTW algorithm. DTW provides a scalar value of the degree of dissimilarity. DTWs are calculated at all combinations of two time series data.

(2) Clustering all grid points

The system subsequently performs clustering of all grid points based on the DTW distances. All grid points are assigned to \( N \) clusters, where the initial value of \( N \) is one. If the inspection for prediction accuracy fails, one is added to \( N \), after which the system performs clustering again.

(3) Constructing prediction models to represent each cluster

The system finally chooses \( N \) grid points representing each cluster. At those points, it performs SVM on a past time series to construct a prediction model every cluster. The SVM furnishes a prediction
After finishing this process, \( N \) prediction models are acquired. To inspect prediction accuracy, the system applies these models to previous solar irradiance data at grid points that belong to each cluster. If the inspection fails, one is added to \( N \), and the system performs clustering again. If the inspection passes, the system applies those models to unlearned current solar irradiance data at grid points belonging to each cluster. Thus, predicted values at multiple points are acquired as output.

### 2.2 Dynamic Time Warping

DTW is a well-known technique to find optimal alignment between two time series data [20,21]. DTW constructs the accumulated cost matrix \( D_{p,p'} \) to calculate the proper distance between two time series with different lengths and cycles. The accumulated cost matrix \( D_{p,p'} = (D_{l,l'}) \) is defined by

\[
D_{l,l'} = \sum_{k=1}^{l'} |a_{p,k} - a_{p',k'}|, \quad D_{1,l'} = \sum_{k=1}^{l'} |a_{p,k} - a_{p',1}|, \quad D_{l,1'} = \sum_{k=1}^{l} |a_{p',k} - a_{p,1}| \quad \text{for} \quad l = 1,...,T, \quad l' = 1,...,T', \quad \text{and}
\]

\[
D_{l,l'} = \min(D_{l-1,l',1}, D_{l-1,l'}, D_{l,l'-1}) + |a_{p,l} - a_{p',l'}| \quad (1)
\]

for \( l \in \{L_p - T, L_p - T'\} \setminus \{2,...,L_p\}, \quad l' = 2,...,L_{p'}, \quad \) where \( a_{p} \) and \( a_{p'} \) are the two time series data. \( T \) is a width of the Sakoe-Chiba band which speeds up DTW computations and prevents an improper alignment [21]. The Sakoe-Chiba band runs along the main diagonal shown as Fig. 2. Alignments can be generated only from white cells. Black cells are not computed innately. The search range is narrowed by appropriately tuning the width \( T \). Fig. 2 conceptually shows how to calculate the accumulated cost matrix \( D_{p,p'} \).

DTW firstly calculates \( D_{1,1} \), the first column \( D_{1,1} \), and row \( D_{1,l'} \). It successively calculates the absolute difference \( |a_{p,j} - a_{p',l'}| \), and then chooses the warping path in order to minimize the current summation \( D_{l,l'} \). At this time, the smallest cell is chosen from the three options \( D_{l-1,l',1}, D_{l-1,l'}, D_{l,l'-1} \) drawn by the arrows in Fig. 2. Eventually, DTW acquires the warping path, and the distance \( d_{DTW}(a_p,a_{p'}) = D_{L_p,L_{p'}} \). This DTW distance represents the degree of dissimilarity.

### 2.3 Hierarchical Clustering

Cluster analysis is a well-known method to divide multiple individuals with variables into clusters based on the degree of dissimilarity between individuals. In particular, hierarchical clustering firstly defines a distance between two clusters. It successively links the closest two clusters until they become only one cluster.

We used the unweighted pair group method with arithmetic mean (UPGMA) for measuring distance between clusters [22]. UPGMA defines an average distance between all individuals belonging to each cluster. The distance between cluster \( C \) and \( C' \) is described by

\[
d_{UPGMA}(C,C') = \frac{1}{|C| \cdot |C'|} \sum_{a_i \in C} \sum_{a_j \in C'} d_{DTW}(a_i,a_j) \quad (2)
\]

where \( |C| \) and \( |C'| \) are the number of elements in \( C \) and \( C' \), respectively. By using this method, the closest two clusters are chosen to be linked. Fig. 3 shows how all clusters are successively linked until they become one cluster. After making the cluster tree, certain \( N \) clusters are acquired by drawing a horizontal line as shown in Fig. 3.

### 2.4 Support Vector Machine

SVM is one of the statistical machine learning methods proposed by Vapnik [23]. It is mainly used in classification and regression. For regression, it is called support vector regression [24,25].

Our system performs SVMs at \( N \) grid points with a mean DTW distance from the northernmost end of each cluster. In this way \( N \) prediction models representing clusters are acquired.

SVM learns a certain \( p \)-th time series data \( a_p = (a_{p,1},...,a_{p,L_p}) \). The learning data are composed of inputs and outputs. This paper adopts third-order auto regression as prediction model for simplification.
The inputs are \( x_{p,m} = (a_{p,m}, a_{p,m+1}, a_{p,m+2})^T \) for \( m = 1, \ldots, n_t \), and the outputs are \( y_{p,m} = a_{p,m+3} \) for \( m = 1, \ldots, n_t \). The prediction model is described by

\[
f_p(x) = w_p^T x + b_p,
\]

where \( w_p = (w_{p,1}, w_{p,2}, w_{p,3})^T \) is a weight vector, \( b_p \) is a bias term, \( f_p(\cdot) \) is a third-order autoregressive model, and \( x \) is a general input third-order vector, composed of three consecutive samples of time series data.

To determine the weight vector, an optimization problem has to be solved. An evaluation function, including both a regularization term and loss term, can be expressed by

\[
\min \frac{1}{2} \| w_p \|^2 + C_p \sum_{m=1}^{n_l} \left| y_{p,m} - f_p(x_{p,m}) \right| \varepsilon_p,
\]

where \( C_p \) is a regularization parameter to keep a balance between the two terms. As the loss term, the \( \varepsilon \)-insensitive loss function is described by

\[
\left| y_{p,m} - f_p(x_{p,m}) \right| \varepsilon_p = \max(0, |y_{p,m} - f_p(x_{p,m})| - \varepsilon_p),
\]

where \( \varepsilon_p \) is a hyperparameter to express an error value to be abandoned. The insensitive loss function suppresses over-fitting and can construct a prediction model of high accuracy.

To solve eq. (4), two slack variables \( \xi_{p,m}, \xi^*_{p,m} \geq 0 \) need to be introduced. The constrained optimization problem is described by

\[
\min \frac{1}{2} \| w_p \|^2 + C_p \sum_{m=1}^{n_l} (\xi_{p,m} + \xi^*_{p,m})
\]

s.t. \( y_{p,m} - f_p(x_{p,m}) \leq \varepsilon_p + \xi_{p,m} \), \( f_p(x_{p,m}) - y_{p,m} \leq \varepsilon_p + \xi^*_{p,m} \).

By the Lagrange multipliers method, the dual problem is described by

\[
\min \frac{1}{2} \sum_{m=1}^{n_l} \sum_{m'=1}^{n_l} (a_{p,m} - a^*_{p,m'}) (a_{p,m} - a^*_{p,m'}) (x_{p,m}^T x_{p,m'}) - \sum_{m=1}^{n_l} (a_{p,m} - a^*_{p,m}) y_{p,m} + \varepsilon_p \sum_{m=1}^{n_l} (a_{p,m} + a^*_{p,m})
\]

s.t. \( \sum_{m=1}^{n_l} (a_{p,m} - a^*_{p,m}) = 0 \), \( 0 \leq a_{p,m} - a^*_{p,m} \leq C_p \) for \( m = 1, \ldots, n_l \).

where \( a_{p,m}, a^*_{p,m} \) are the Lagrange multipliers. Among solutions, learning data with \( a_{p,m}, a^*_{p,m} > 0 \) are support vectors. A prediction model can finally be given by

\[
f_p(x) = \sum_{m=1}^{n_l} (a_{p,m} - a^*_{p,m}) x_{p,m}^T x + b_p.
\]

Under \( \xi_{p,m}, \xi^*_{p,m} = 0 \), considering the constraint conditions of eq. (6), a bias term is described by

\[
b_p = \begin{cases} y_{p,m} - \sum_{m'=1}^{n_l} (a_{p,m'} - a^*_{p,m'}) x_{p,m'}^T x_{p,m} - \varepsilon_p, & 0 < a_{p,m'} < C_p, \\ y_{p,m} - \sum_{m'=1}^{n_l} (a_{p,m'} - a^*_{p,m'}) x_{p,m'}^T x_{p,m} + \varepsilon_p, & 0 < a^*_{p,m'} < C_p. \end{cases}
\]

The performance is improved by appropriately tuning hyperparameters \( C_p \) and \( \varepsilon_p \). In this work the hyperparameters are tuned by using a well-known grid search for simplification. We firstly select a finite set of those parameters. SVMs are learned per pair, and then the pair with the highest performance is chosen. Tentatively the set was \( C_p = \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2, 10^3, 10^4\} \) and \( \varepsilon_p = \{0, 5, 10, 15, 20, 25, 30, 35, 40\} \) in this research.

### 2.5 Prediction Accuracy

Prediction accuracy is evaluated by root mean square error (RMSE), which is one of the most popular indicators to evaluate prediction accuracy [3,6]. It is expressed by

\[
\text{RMSE} = \sqrt{\frac{1}{n_t} \sum_{t=1}^{n_t} \left( y_{p,t} - f_p'(x_{p,t}) \right)^2},
\]

where \( f_p'(x_{p,t}) \) and \( y_{p,t} \) are predicted and true values, respectively. RMSE basically indicates the degree of error; when its value is small, prediction accuracy is high.

### 3. Simulation

#### 3.1 Data Description

##### 3.1.1 Observational Satellite Data

We used global solar irradiance data [W/m²] provided by AMATERASS as shown in Fig. 4 [18]. AMATERASS is able to acquire solar irradiance data of 1-km spatial resolution and 30-min temporal resolution over Japan. Fig. 5 shows an example of half-hourly time series data for two weeks. It has approximately 16 hours per day, excluding the nighttime hours. In total, it has approximately 448 (=16 hours × 2 × 14 days) samples. In this paper, half-hourly time series data at multiple points were tentatively used for the aforementioned time series data \( a_p \). The entire period of analysis was almost one year from November 2013 to August 2014.

##### 3.1.2 Japan Mesh Model

As an initial analysis, Japan mesh models (Fig. 6) were introduced that generated multiple points across the country. We made the models based on the grid square code constructed by Japanese Industrial Standards [26]. Experimental targets were 102 and 408 lattice points. On the mesh model of 102 grid points, latitude and longitude differences between two points are \( 40' \) and \( 1' \), respectively. The length of a side is approximately 80 km. On one of the 408 grid points, latitude and longitude differences between two points are \( 20' \) and \( 0.5' \), respectively. The length of a side is approximately 40 km. For convenience, each grid point was assigned a number indicating its location. The
3.2 Results

3.2.1 Measuring Similarities of Time Series Data

Fig. 7 presents the results of the similarities of time series data that were measured in January. In this paper, the DTWs were computed for time series data for two weeks with the Sakoe-Chiba bandwidth $T = 15$. The figure is expressed in matrix form; the row and column are the indices of the grid points, and the cells are the DTW distances between two grid points of the row and column. The matrix is needed for clustering grid points in the next process in 3.2.2. When performing clustering, the two grid points of the row and column with the smallest cell in the matrix are chosen to be linked. The linked cluster is added to the row and column instead of the two grid points. This procedure is repeated until all the grid points become one cell.

We compared the calculation time for 102 grid points and that for 408 grid points. The computer specifications were: Windows 7, Intel® Core™ i7-5960X CPU @ 3.00 GHz, 64.0 GB RAM. MATLAB R2016a was used for numerical analysis. In this case, 173 seconds were required for 102 grid points and 2768 seconds for 408 grid points. When the spatial resolution increased by twice the amount, the number of grid points increased by a factor of four and the required time for completion was 16 times greater. Therefore, the square of the rate of increase in the number of grid points corresponded to the rate of increase in calculation time. This was the reason DTWs were calculated based on combinations of two grid points selected from all grid points.

3.2.2 Clustering All Grid Points

Figs. 8–11 show the results of clustering grid points based on DTW distances for $N = 2$ of 102 grid points in November, $N = 6$ of 102 grid points in April, $N = 2$ of 408 grid points in November, and $N = 6$ of 408 grid points in April, respectively. Even though the cluster analysis did not use location information of the grid points, the geographical characteristics of Japan were clear. These results proved that the time series solar irradiance data had the geographical properties of Japan.

As $N$ increased, it was obvious that the clusters revealed the geographical characteristics in detail. When $N = 2$ of 102 grid points in November, Cluster 1 roughly represented the north part and Cluster 2 the south part as shown in Fig. 8. When $N = 6$ of 102 grid points in April, Cluster 1 represented the Hokkaido, and Cluster 2 the Tohoku and Hokuriku. Cluster 3 characteristically had two separate regions: Kanto, and Chugoku and Shikoku. Cluster 4 faced the Pacific Ocean, Cluster 5 represented Kansai, and Cluster 6 was the Kyushu, as shown in Fig. 9. These clusters nearly corresponded to the climatic division depend-
ing on the cluster numbers and seasons. This means DTW-based clustering can flexibly represent the climatic division with respect to solar irradiance. We also clustered 408 grid points as shown in Figs. 10, 11. Compared to the results obtained for 102 grid points, the boundaries of the cluster were much clearer. When $N = 6$, the central part, especially around Cluster 5, differed from the results obtained with 102 grid points. This result indicated that the central part was obscured to a certain degree and not easy to clearly cluster.

We identified the clustering calculation time. Based on the aforementioned computer specifications, one second or less was required, because we already calculated DTWs. This is an extremely short amount of time compared to the time required by DTWs or SVMs.

### 3.2.3 Constructing Prediction Models to Represent Each Cluster and Inspecting Prediction Accuracy

Figs. 12–15 show the results of predicting solar irradiance using models to represent each cluster. These figures present mean RMSEs of predicting all grid points for variable $N$ and naïve method. The naïve method is the simplest prediction model and provides a benchmark. In the naïve method, the predicted value equals the last observed value as $\hat{a}_{p,m+1} = a_{p,m}$. Moreover, when $N = 102$ (drawn by squares), prediction models are normally constructed at all grid points without using the proposed system.

The proposed system requires as low $N$ as possible to reduce SVM computation while maintaining high prediction accuracy. Further, when the season changes, a new $N$ is required for precise prediction. For these reasons, an automatic tuning of $N$ is necessary for the proposed system. This system automatically tunes $N$ by inspecting the prediction accuracy one by one starting from $N = 1$, using previous data. Considering that we cannot know the accuracy in the future after $N + 1$, we devised a method for tuning $N$. Current $N$ passes the inspection if both of the following two conditions are satisfied.

- Mean RMSE of all grid points with current $N$ is less than that with the naïve method.
- Mean RMSE of all grid points with current $N$
is less than that with $N - 1$ by $q$ or more.

The first condition requires the prediction accuracy with $N$ to be higher than with the naive method as a benchmark. The second condition compares the current accuracy with that previously obtained. $q$ is a rate of change of RMSE, and a parameter to control the balance between the accuracy and calculation time. When $q$ is set higher, $N$ is expected to be tuned to a greater extent; consequently, the accuracy will be higher. $q$ is tentatively one in this research. Furthermore, we need to add another condition in case the aforementioned conditions remain unsatisfied. Considering calculation time, the upper bound of $N$ could tentatively be ten. Namely, another condition is that the current $N$ passes the inspection when it becomes ten. If the system followed the aforementioned conditions, the values of $N$ for the four seasons would be tuned as shown by circles in Figs. 12–15. The RMSE values obtained for the four seasons with $N = N_f$ were less than those obtained with the naive method, and they were not much more than those obtained with $N = 102$. Therefore, we selected low values for $N$ while maintaining high prediction accuracy in our study.

We performed a posteriori analysis by continuing the inspection until $N$ becomes ten, as shown by dots in Figs. 12–15. The RMSE values obtained for the four seasons with $N = N_f$ were less than those obtained with the naive method, and they were not much more than those obtained with $N = 102$. Therefore, we selected low values for $N$ while maintaining high prediction accuracy in our study.

4. Conclusions

This paper introduced a multi-point prediction system that reduced the amount of calculation. In addition, we solved the problems related to realizing a practical system: further reducing the amount
of calculation, managing an increase in the number of grid points, and automating an algorithm. Results revealed that the system saved a considerable amount of calculation time while maintaining high prediction accuracy. However, a problem remains with respect to updating the prediction models. In fact, automatically finding the proper timing for updating models is difficult because we cannot know the future prediction accuracy. An effective method of updating the model is necessary to finalize the system.

Acknowledgements

This work was supported in part by Japan Science and Technology Agency, CREST.

References

[1] Y. Tanaka and M. Takahashi: Dynamic time warping-based cluster analysis and support vector machine-based prediction of solar irradiance at multipoints in a wide area; Proceedings of the 47th ISCIE International Symposium on Stochastic Systems Theory and Its Applications, pp. 210–215 (2016)

[2] Z. Dong, D. Yang, T. Reindl and W. M. Walsh: A novel hybrid approach based on self-organizing maps, support vector regression and numerically predicted cloudiness to forecast solar irradiance; Energy, Vol. 82, pp. 570–577 (2015)

[3] K. Benmouiza and A. Cheknane: Forecasting hourly global solar radiation using hybrid k-means and non-linear autoencoder neural network models; Energy Conversion and Management, Vol. 75, pp. 561–569 (2013)

[4] A. Sfetsos and A. H. Coonick: Univariate and multivariate forecasting of hourly solar radiation with artificial intelligence techniques; Solar Energy, Vol. 68, No. 2, pp. 169–178 (2000)

[5] G. Reikard: Predicting solar radiation at high resolutions: A comparison of time series forecasts; Solar Energy, Vol. 83, pp. 342–349 (2009)

[6] K. Mohammadi, S. Shamshirband, M. H. Anisi, K. A. Alam and D. Petkovic: Support vector regression based prediction of global solar radiation on a horizontal surface; Energy Conversion and Management, Vol. 91, pp. 433–441 (2015)

[7] K. Kawamoto, T. Nakajima and T. Y. Nakajima: Estimation of solar radiation using a neural network based on radiative transfer; Journal of Geophysical Research, Vol. 116, DO8215, doi: 10.1029/2010JD014337 (2011)

[8] H. Takenaka, T. Y. Nakajima, A. Higurashi, A. Higuchi, T. Takamura, R. T. Pinker and T. Nakajima: Estimation of solar radiation using a neural network based on radiative transfer; Journal of Geophysical Research, Vol. 116, DO8215, doi: 10.1029/2010JD014337 (2011)

[9] M. Muller: Information Retrieval for Music and Mo-
Authors

Yukiya Tanaka

He received B.E. degree from the Department of System Design Engineering, Keio University, Yokohama, Japan in 2015. His research interests are machine learning, time series analysis, and predictive control.

Masaki Takahashi (Member)

He received Ph.D. degree in Engineering from the School of Science for Open and Environmental Systems, Keio University, Yokohama, Japan in 2004. From 2005 to 2008, he worked as a Research Assistant in the Department of System Design Engineering, Keio University, and became an Associate Professor in 2009. His research interests include human-robot interaction, motion and vibration control, and sensor fusion. He is a member of The American Institute of Aeronautics and Astronautics, The Japan Society of Mechanical Engineers, The Robotics Society of Japan, and The Institute of Systems, Control and Information Engineers.