Development of an efficient real-time disruption predictor from scratch on JET and implications for ITER

2013 Nucl. Fusion 53 113001

(http://iopscience.iop.org/0029-5515/53/11/113001)

IOP PUBLISHING and INTERNATIONAL ATOMIC ENERGY AGENCY NUCLEAR FUSION

Nucl. Fusion **53** (2013) 113001 (8pp) doi:10.1088/0029-5515/53/11/113001

Development of an efficient real-time disruption predictor from scratch on JET and implications for ITER

S. Dormido-Canto[1], J. Vega[2], J.M. Ramírez[1], A. Murari[3], R. Moreno[2], J.M. López[4], A. Pereira[2] and JET-EFDA Contributors[a]

JET-EFDA Culham Science Centre, Abingdon, OX14 3DB, UK
[1] Dpto. Informática y Automática-UNED, Madrid, Spain
[2] Asociación EURATOM/CIEMAT para Fusión, Madrid, Spain
[3] Associazione EURATOM/ENEA per la Fusione, Consorzio RFX, 4-35127 Padova, Italy
[4] Grupo de Investigación en Intrumentación y Acústica Aplicada, UPM, Madrid, Spain

Received 3 June 2013, accepted for publication 16 August 2013
Published 11 September 2013
Online at stacks.iop.org/NF/53/113001

Abstract
Prediction of disruptions from scratch is an ITER-relevant topic. The first operations with the new ITER-like wall constitute a good opportunity to test the development of new predictors from scratch and the related methodologies. These methodologies have been based on the Advanced Predictor Of DISruptions (APODIS) architecture. APODIS is a real-time disruption predictor that is in operation in the JET real-time network. Balanced and unbalanced datasets are used to develop real-time predictors from scratch. The discharges are used in chronological order. Also, different criteria to decide when to re-train a predictor are discussed. The best results are obtained by applying a hybrid method (balanced/unbalanced datasets) for training and with the criterion of re-training after every missed alarm. The predictors are tested off-line with all the discharges (disruptive/non-disruptive) corresponding to the first three JET ITER-like wall campaigns. The results give a success rate of 93.8% and a false alarm rate of 2.8%. It should be considered that these results are obtained from models trained with no more than 42 disruptive discharges.

1. Introduction

During tokamak operations disruptive events, e.g. disruptions, can take place. They occur due to loss of stability and/or confinement, which cause the abrupt termination of the discharge [1, 2]. In large fusion devices, such as JET, it is not uncommon for disruptions to constitute a considerable risk for the structural integrity of the machine [3]. In JET, the number of disruptions varies between campaigns [4]. It should be noted that in JET campaigns with a metallic wall (C28–C30, from July 2011 to June 2012), the number of unintentional disruptive discharges has increased compared with previous campaigns [5]. Therefore, in this context, disruptions are one of the crucial research issues in the short term in view of ITER.

The physical characterization of disruptions for classification, prediction and control is an extremely complex task. The large number of plasma parameters and the complex interaction between the plasma and external tokamak systems makes development of a first-principles physical model to predict disruptions almost impossible. On the other hand, prediction is essential to implement mitigation actions. To overcome the lack of theoretical basis for disruptions, several machine learning techniques, mainly artificial neural networks and support vector machines (SVM), have been used as an alternative approach to disruption prediction [6–11]. These computational systems are capable of building general models that 'learn' from the data in the so-called 'training process'. In all of the above works, the training processes had access to the whole databases of the respective fusion devices and, therefore, thousands of discharges were available to accomplish the learning procedure. Once the systems were trained, the models could be used to detect the presence of an incoming disruption. Therefore, two important facts have to be emphasized. On the one hand, the disruption predictors have to work in real time (RT) and have to provide high success rates simultaneously with low rates of false and missed alarms. On the other hand, a valuable parameter to characterize the predictors is how early in advance the disruptions are identified.

Focusing the attention on these points, it should be noted that the respective papers on the predictors [6, 7, 9], do not

[a] See the appendix of Romanelli F. *et al* 2012 *Proc. 24th IAEA Fusion Energy Conf. (San Diego, CA, 2012)* www-pub.iaea.org/iaeameetings/41985/24th-Fusion-Energy-Conference.

mention RT capabilities. In the case of [8], the proposed neural predictor is suitable for RT application but no RT use has been reported. The predictor [10] is tested in a simulated RT environment. Finally, an Advanced Predictor Of DISruptions (APODIS) [11] is in operation in the JET RT data network. It is not based on a single RT signal but on a parameter space of seven RT signals: plasma current, mode lock amplitude, total input power, plasma internal inductance, plasma density, stored diamagnetic energy time derivative and radiated power. Its results during the first three ITER-like wall (ILW) campaigns (C28–C30) in JET are summarized by the following statistics: success rate 98.36%, false alarms 0.92% and missed alarms 1.64%. With regard to the warning time, the APODIS predictor triggers the alarms in JET 426 ms (in average) before the disruptions.

From the ITER viewpoint, the prediction of disruptions is an essential problem to be solved. For this purpose, it should be clear that automatic machine learning methods can be used, but it is important to point out the big difference with present fusion devices: ITER needs reliable disruption predictors without having to wait for hundreds of disruptions. Therefore, the development of disruption prediction systems that achieve high learning rates with small training datasets is crucial. Moreover, one should try to understand how fast new predictors can be trained. In relation to this and taking into account the necessary learning process, one of the main problems is the selection of a good enough set of discharges to achieve 'relevant results' together with the important constraint of a limited amount of information. The term 'relevant results' means that the predictions have to show high success rates simultaneously with minimum rates of both false alarms and missed alarms. In other words, searching for good enough disruption predictors implies to solve the multi-objective problem of RT identification, early prediction to be able to apply mitigation actions, high success rates, low false alarms and low missed alarms.

This paper is focused on the development of a methodology to create disruption predictors under ITER-relevant conditions. The ITER-relevant conditions impose to solve the above multi-objective problem starting from scratch, without any previous knowledge about disruptions. These kinds of disruption predictors are new in nuclear fusion and their robustness and reliability have to be assessed. In order to achieve this aim, the methodology has been applied to the JET database, in particular to the ILW campaigns, as starting JET operation with a metallic wall can be considered the closest situation to start the operation of a new fusion device.

With regard to the structure of the paper, section 2 is devoted to introducing the proposed methodology by using a simplified example. Section 3 describes the implementation framework with a brief introduction to APODIS and the database used. Section 4 reports the results obtained from the metallic wall JET campaigns. A final discussion with the prospects for further investigations is the subject of the last section.

2. Methodology

This section describes proposals of methodologies to develop disruption predictors from scratch. In this work, the methodologies are particularized for binary classification tasks; however, they can be considered quite generic and valid for multi-class problems. The objective is to create a predictor from scratch to be able to classify (at any time of a discharge) the plasma behaviour as disruptive or non-disruptive.

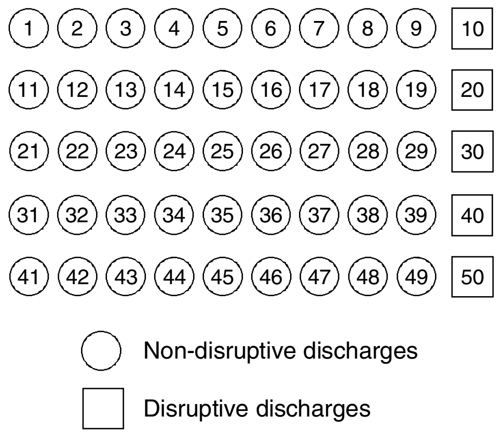

Figure 1. Simplified example.

In any tokamak, the number of non-disruptive discharges, compared with the number of disruptive discharges, is very high. In this sense, the fusion databases to apply automatic learning techniques for disruption prediction are highly unbalanced. Moreover, it should be noted that in any tokamak, such as ITER, the unbalance between both types of discharges is an intrinsic property. Therefore, from a methodological point of view, the prediction of disruptions from scratch requires an in depth analysis to determine the influence of this fact on the training process. Also, it is important to note that disruption predictors from scratch have to be generated as discharges are produced. This means the use of the discharges in chronological order as they occur. This is a realistic way of developing predictors, thereby allowing the development of adaptive predictors that are continuously learning. In the case of new devices (for instance ITER) or new operation conditions (for example, the ILW operation in JET), the adaptive predictors can learn from scratch following the plasma evolution from moderate density and temperature conditions to high-performance scenarios.

Regardless of the use of balanced or unbalanced approaches, an important decision in the methodology is to establish when re-training is required to obtain a new predictor. In other words, it is necessary to define when new knowledge has to be incorporated to learn something new from the data. A first criterion has been to generate a new predictor just after the occurrence of every new disruptive discharge. The reason for this is to incorporate any new information from disruptive discharges to be used in the future prediction of disruptions.

To explain this criterion, let us consider the simplified example of figure 1, where a set of 50 discharges (in chronological order) is represented. The example considers 45 non-disruptive discharges (represented by circles) and 5 disruptive discharges (denoted by squares). It is just an illustrative example and therefore, for didactic purposes, the disruptions are chosen evenly spaced. From a practical point of view, the methodology is exactly the same regardless of the number of disruptive discharges or the variable number of

Table 1. Training models with unbalanced datasets for the simplified example of figure 1.

Training model	Discharges	i	NDD(i)
$M(1)$	*DD*: {10} *NDD*: {[1, 9]}	1	9
$M(2)$	*DD*:{10, 20} *NDD*: {[1, 9] ∪ [11,19]}	2	18
$M(3)$	*DD*: {10, 20, 30} *NDD*: {[1, 9] ∪ [11, 19] ∪ [21, 29]}	3	27
$M(4)$	*DD*: {10, 20, 30, 40} *NDD*: {[1, 9] ∪ [11, 19] ∪ [21, 29] ∪ [31, 39]}	4	36
$M(5)$	*DD*: {10, 20, 30, 40, 50} *NDD*: {[1, 9] ∪ [11, 19] ∪ [21, 29] ∪ [31, 39] ∪ [41, 49]}	5	45

non-disruptive discharges between disruptive ones. A greater number of disruptive discharges will result in a larger number of trained models. According to the criteria of generating a predictor after every disruptive discharge, the situation simulated in figure 1 requires the generation of five predictors.

2.1. Unbalanced training datasets

In binary classification problems, unbalanced training datasets are particular cases of automatic machine learning methods that have to be developed when there are significantly fewer training instances of one class compared with the other class. It is well known that in the case of unbalanced datasets, the decision boundary established by standard machine learning algorithms tends to be biased towards the majority class; therefore, the minority class instances show a certain risk of misclassification. As mentioned before, in the normal operating scenario of tokamaks, the unbalance between disruptive and non-disruptive discharges is an intrinsic property. This means that the number of non-disruptive discharges (majority class) is much higher than disruptive discharges (minority class). The degree of unbalance can be represented by the ratio of the sample size of the minority class to that of the majority class. In our context (disruptions in JET), it is possible to assume that the ratio of minority to majority samples can be approximately 1 to 10. During the last period of carbon wall operation the unbalance was smaller (4% disruptivity) [12]. On the other hand, during the first campaigns with the ILW, the percentage of disruptions increased significantly [5]. Therefore, in the simplified example (figure 1) the degree of unbalance considered is 1 to 10.

The notation used to identify the trained classifiers (models) can be summarized as follows:

- d is the number of disruptive discharges (*DD*).
- $d(i)$ is the ordinal position for the ith disruptive discharge in the sequence of all discharges (in the simplified example for $i = 1 \Rightarrow d(1) = 10$, for $i = 2 \Rightarrow d(2) = 20$ and so on).
- $M(i)$ is the model trained when the ith disruptive discharge occurs (in the simplified example $i = 1, 2, \ldots, 5$).
- $nd(i)$ is the ordinal position for the ith non-disruptive discharge in the sequence of all discharges (in the simplified example for $i = 1 \Rightarrow nd(1) = 1$, for $i = 18 \Rightarrow nd(18) = 19$ and so on).
- *NDD*(i) is the number of non-disruptive discharges (*NDD*) in the model $M(i)$. For example, in the simplified example of figure 1, $M(4)$ corresponds to a model made up of four disruptive discharges (numbers 10, 20, 30 and 40) and *NDD*(4)= 36 non-disruptive discharges.

Details of the discharge selections for training models with unbalanced datasets are summarized in the following pseudocode:

$d(0) = 0;$
$nd(0) = 1;$
for $i = 1 : d$

$$M(i) \subset \left\{ \begin{array}{l} DD: \bigcup_{k=1}^{i} d(k) \\ NDD: \bigcup_{k=1}^{i} \{[nd(d(k-1)+1), nd(d(k)-i)]\} \end{array} \right\}$$

end for

Table 1 shows the five models of training with unbalanced datasets for the simplified example of figure 1. The first column represents the training model. The second one is the shot numbers belonging to each training model split into disruptive discharges (*DD*) and non-disruptive discharges (*NDD*). The third column shows the number of disruptive discharges (matched with the training model) and the last one reports the number of non-disruptive discharges.

2.2. Balanced training datasets

In binary classification problems, one talks of balanced training datasets when there are more or less the same numbers of instances of both classes. In this sense, the balance can be seen as a guarantee of fairness. Most machine learning algorithms work well with balanced datasets since they aim to optimize the overall classification accuracy. In this case, the degree of unbalance is 1 to 1. However, it should be noted that the unbalance ratio between classes is not the only factor that can reduce classification performance; other factors such as training size and complexity can also affect the performance. The complexity corresponds to the level of separability of classes within the data. For datasets that are linearly separable, classifier performances are not susceptible to any amount of unbalance [13]. As the degree of data complexity increases, the class unbalance factor starts influencing the classifier generalization ability. In the present adaptive predictor, as the unbalance increases, the complexity is greater. As the number of non-disruptive discharges is

Table 2. Training models with balanced datasets for the simplified example.

Training model	Discharges	i	NDD(i)
$M(1)$	*DD*: {10} *NDD*: 1-random from {[1, 9]}	1	1
$M(2)$	*DD*:{10, 20} *NDD*: 2-random from {[1,9] ∪ [11, 19]}	2	2
$M(3)$	*DD*: {10, 20, 30} *NDD*: 3-random from {[1, 9] ∪ [11, 19] ∪ [21, 29]}	3	3
$M(4)$	*DD*: {10, 20, 30, 40} *NDD*: 4-random from {[1, 9] ∪ [11, 19] ∪ [21, 29] ∪ [31, 39]}	4	4
$M(5)$	*DD*: {10, 20, 30, 40, 50} *NDD*: 5-random from {[1, 9] ∪ [11, 19] ∪ [21, 29] ∪ [31, 39] ∪ [41, 49]}	5	5

significantly higher than the number of disruptive discharges, the balanced training approach requires a random selection of non-disruptive discharges.

Details of the discharge selection for training with balanced datasets are summarized in the following pseudocode:

$d(0) = 0;$
$nd(0) = 1;$
for $i = 1 : d$

$$M(i) \subset \left\{ \begin{array}{l} DD : \bigcup_{k=1}^{i} d(k) \\ i \text{ random } NDD \text{ from: } \bigcup_{k=1}^{i} \{[nd(d(k-1)+1), \\ \qquad nd(d(k)-i)]\} \end{array} \right\}$$

end for

Table 2 shows the five models of training with balanced datasets for the simplified example of figure 1. The first column represents the model. The second one reports the shot numbers belonging to each training model that are split into disruptive discharges (*DD*) and non-disruptive discharges (*NDD*). The third column shows the number of disruptive discharges and the last one reports the number of non-disruptive discharges. In this case, the number of disruptive and non-disruptive discharges is the same.

2.3. Test of models

The testing phase of each model is performed with all the discharges produced after the ones used for training. Details of the discharge selection to test each trained model are summarized in the following pseudocode:

for $i = 1 : d$
$M(i)$ is tested with $[d(i) + 1, end]$
end for

where *end* represents the total number of discharges.

Table 3 shows the test sets used for each trained model in the simplified example of figure 1.

Evaluation metrics play an important role in machine learning. They are used to assess and guide the learning algorithms. In the case of unbalanced datasets, if a particular metric is chosen and it does not properly evaluate the minority class, then the learning algorithms will not be able to efficiently handle the unbalanced problem.

A typical metric that is quite common in machine learning is the overall classification rate (i.e. accuracy). However, on an unbalanced dataset, the overall classification is no longer a suitable metric, since the small class has less effect on accuracy as compared with the prevalent class [14]. In our case, a multi-objective complex optimization problem is considered: the RT achievement of high success rates in the disruption predictions and simultaneously the highest reduction of the false alarm rates, i.e.

Table 3. Test sets for the simplified example of figure 1.

Training model	Test sets
$M(1)$	{[11, 50]}
$M(2)$	{[21, 50]}
$M(3)$	{[31, 50]}
$M(4)$	{[41, 50]}
$M(5)$	———

$$\max \left\{ \frac{\text{Disruption predictions}}{\text{Disruptive discharges}} \right\} \text{ and } \times \min \left\{ \frac{\text{Disruption predictions}}{\text{Non-disruptive discharges}} \right\}$$

where max and min represent the maximum and minimum functions, respectively.

Therefore, the following two extreme cases can be considered:

- *Case 1.* The occurrence of disruptions is predicted for all shots. A prediction success rate of 100% can be obtained with a false alarm rate close to 100%.
- *Case 2.* The occurrence of disruptions is not predicted for any shots. A false alarm rate of 0% can be obtained with a prediction success rate of 0%.

The prediction performance can be shown on a diagram plotting the prediction success rate versus the false alarm rate (see figure 2). This diagram shows that an overall high prediction performance, which means a positive prediction success rate of ∼100% and a low false alarm rate of ∼0% at the same time, should be realized. It is meaningless to pursue high prediction success rates only.

3. Implementation framework

3.1. APODIS

The proposed methodologies to train from scratch have been implemented with the APODIS. At present, APODIS is running in the JET RT data network [15] on a routine basis

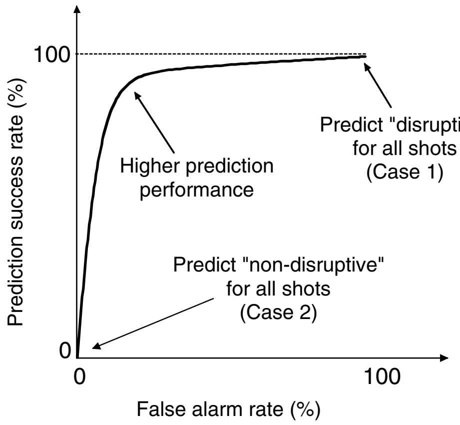

Figure 2. Prediction performance shown by plotting the prediction success rate against false alarm rate.

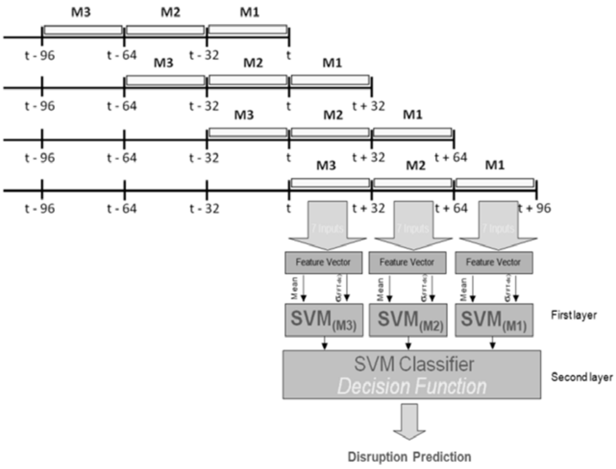

Figure 3. APODIS architecture. The first layer is formed by three radial basis function (RBF) kernel SVM classifiers and the second layer is a linear kernel classifier.

within the Multithreaded Application Real-Time executor (MARTe) framework [16]. Figure 3 shows the architecture of APODIS based on two layers of SVM classifiers. The first layer follows the temporal evolution of the plasma through three sequential SVM classifiers (M1, M2 and M3, based on RBF kernels, where M1 analyses the last 32 ms of each discharge, M2 the previous 32 ms interval and M3 the 32 ms interval before M2) and the second one implements a different SVM classifier (linear kernel) to trigger or not an alarm.

APODIS can be configured to use different sets of relevant signals depending on the needs. These signals are processed using 32 ms time windows with a sampling frequency of 1 kHz. For each time window, two features are calculated per signal: (1) the mean value and (2) the standard deviation of the fast Fourier transform (FFT) removing the zero frequency component.

For the training models based on APODIS, it has been necessary to use high-performance computing (HPC) available at CIEMAT. This HPC environment consists of 240 nodes of 2 Quad-Core Xeon processors (X5450 and X5570) at 3 GHz with 16 GB of RAM memory.

Table 4. Database of C28–C30 JET campaigns.

	Number of discharges	Range of pulses
Disruptive	201	[81852, 83793]
Non-disruptive	1036	
Unbalanced degree	≅ 1 to 5	

Table 5. List of signals.

Id number	Signal name
(1)	Plasma current
(2)	Mode lock amplitude
(3)	Plasma internal inductance
(4)	Plasma density
(5)	Stored diamagnetic energy time derivative
(6)	Radiated power
(7)	Total input power
(8)	Poloidal beta
(9)	Plasma vertical centroid position
(10)	Plasma internal inductance time derivative
(11)	Poloidal beta time derivative
(12)	Plasma vertical centroid position time derivative

3.2. The database

In order to demonstrate the robustness and generalization capabilities of the disruption prediction methodologies summarized in sections 2.1 and 2.2, a database from JET ILW campaigns is selected. Table 4 summarizes the number and the range of pulses that are used. Shots with incomplete or unreliable measurements are excluded. For prediction purposes, only natural disruptions are used to train the models.

On JET, thousands of signals are acquired in every pulse. The selection of the most informative physical quantities is fundamental to properly identify an incoming disruption. On the one hand, too many signals could overload the learning capacity of an automatic system. On the other hand, too few could not provide enough information to perform reliable predictions. Twelve signals are chosen for training and testing models (see table 5). The majority or all of these signals have also been used in previous research on disruptions with quite good results [8–10, 17, 18]. In this paper, two signal sets are considered: an expanded set (that includes all the signals in table 5) and a reduced set (that includes the signals from (1) to (7)). The latter is the set of signals used by APODIS in the JET RT data network.

4. Results

The performances of the models trained from scratch with the approaches shown in section 2 are compared. Figure 4 displays the disruption prediction for the unbalanced dataset approach with both the expanded signal set (represented by black lines) and the reduced signal set (represented by grey lines). The x-axis represents the number of disruptive discharges in each model. For example, 106 means that the model 106 is made up of the first 106 disruptive discharges (in chronological order) and all the non-disruptive discharges that have occurred so far (in the present case, exactly 678 non-disruptive discharges). Each classifier is evaluated with the rest of the discharges produced after it is generated. For classifier 106, the test set

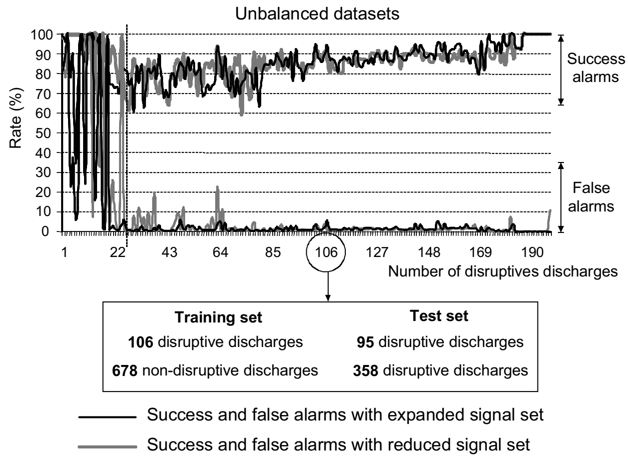

Figure 4. Disruption prediction results from scratch with unbalanced datasets.

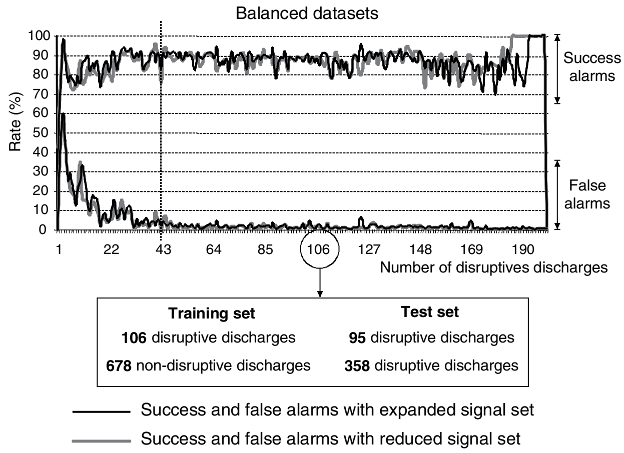

Figure 5. Disruption prediction results from scratch with balanced datasets.

Table 6. Average success and false alarm rates with the standard deviation for unbalanced datasets with expanded and reduced signal sets. Data are computed from model 24.

	Success rate (%)	False alarm rate (%)
Expanded signal set	85.99 ± 8.43	1.37 ± 1.19
Reduced signal set	85.65 ± 7.78	2.27 ± 3.06

Table 7. Average success and false alarm rates with the standard deviation for balanced datasets with expanded and reduced signal sets. Data are computed from model 42 onwards.

	Success rate (%)	False alarm rate (%)
Expanded signal set	87.63 ± 5.27	1.76 ± 1.01
Reduced signal set	88.80 ± 5.41	1.48 ± 0.98

is therefore formed by 95 disruptive discharges and 358 non-disruptive discharges.

It is important to note the improvement of results (both greater success rate and lesser false alarm rate) as new discharges are incorporated to the training set. Until model 24, the predictions are quite erratic and unstable. The interpretation for this is that the predictor does not incorporate enough knowledge and more training is required. Model 24 determines a clear frontier and the models beyond this point show better results. The positive trend in the success rate together with the decrease in the false alarm rate confirms the better behaviour of the predictors. Table 6 shows the average rates of success and false alarms with the standard deviation from model 24 onwards with both expanded and reduced signal sets.

The standard deviation in the false alarms with the reduced signal set is slightly high because there are some peaks in the first models, as shown in figure 4.

Figure 5 shows the disruption prediction for the balanced dataset approach with both the expanded signal set (represented by black lines) and the reduced signal set (represented by grey lines). The x-axis gives the number of disruptive discharges in each model. For example, 106 represents model 106, which is made up of the first 106 disruptive discharges (in chronological order) and 106 non-disruptive discharges randomly chosen from the set of non-disruptive discharges that have occurred so far (in this case, 106 non-disruptive discharges are chosen from a total number of 678). As in the unbalanced case, each classifier is evaluated with the rest of the discharges produced after it is generated. For classifier 106, the test set therefore consists of 95 disruptive discharges and 358 non-disruptive discharges.

Unlike what happened with the unbalanced datasets, the present approach obtains better results from the first models in terms of stability, that is, the number of oscillations has significantly been reduced. It should be noted that results are very stable (in terms of success and false alarm rates) from model 42. Table 7 shows the average rates of success and false alarms with the standard deviation from model 42 onwards with both expanded and reduced signal sets.

After the analysis of the two previous approaches, a question arises: which approach should be considered to develop reliable predictors? To answer this, it is important to look at tables 6 and 7. Unbalanced datasets show lower false alarms and success rates than the balanced approach. Looking for simultaneously satisfying a high success rate (as the one obtained with balanced datasets) and a low false alarm rate (as in the case with unbalanced datasets), a hybrid approach can be implemented. The hybrid approach is based on starting the generation of predictors with balanced datasets (to avoid the unstable and erratic predictions of unbalanced datasets) and switching to unbalanced trainings at a certain point with the aim of maintaining the false alarms as low as possible. The switching point establishes the number of non-disruptive discharges to be used in the training datasets. From that point onwards, the training datasets will be unbalanced, always with the same number of non-disruptive discharges (randomly chosen from the database) and an increasing number of disruptive discharges (each new disruption is included in the new training datasets).

According to figure 5, after model 42, the results are quite stable and this is the number of non-disruptive discharges to be used in a hybrid approach. Figure 6 plots the success and false alarm rates with the hybrid approach and shows more steady results. The predictors are trained with balanced datasets up to the disruption number 42. From that moment, the training will be unbalanced in favour of the disruptive discharges. Table 8 shows the average rates of success and false alarms with the

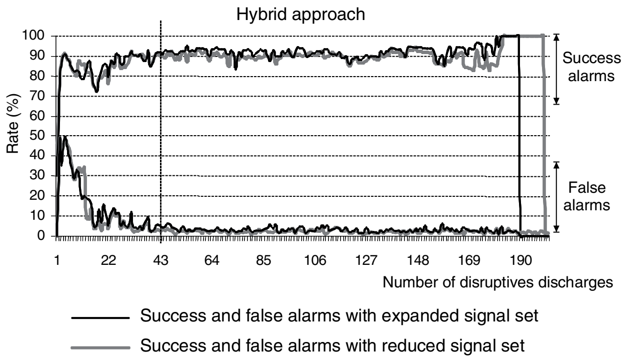

Figure 6. Disruption prediction results from scratch with a hybrid approach.

Table 8. Average success and false alarm rates with the standard deviation for a hybrid approach with both expanded and reduced signal sets. Data are computed from model 42.

	Success rate (%)	False alarm rate (%)
Expanded signal set	93.03 ± 2.91	3.14 ± 1.14
Reduced signal set	91.27 ± 3.99	2.25 ± 0.80

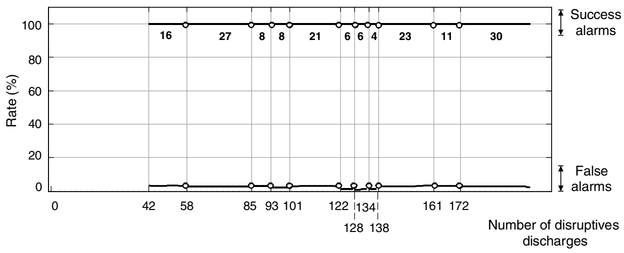

Figure 7. Disruption prediction results from scratch with a hybrid approach. Re-trains are carried out after missing a disruption, which happens at the disruptions 58, 85, 93 and so on.

standard deviation of the hybrid approach from model 42 with both expanded and reduced signal sets.

With regard to figure 6, the sharp fall in the success rate at the right extreme of the plot should be properly interpreted. It is clearly derived from the fact that the last models use very reduced test sets. For example, for model 200, there is only one disruptive discharge in the test set and, therefore, if it is missed, the success rate falls to 0%.

To conclude, it is important to discuss whether the creation of new predictors is really necessary each time a new disruption occurs. If a disruption happens and it has been predicted, a new training can be avoided because the automatic learning system is able to recognize the incoming disruption. Therefore, no new knowledge has to be incorporated to the predictor. In the case of missing a disruption, a re-training with the current disruption data is required to include in the predictor new information not available before.

Figure 7 displays the success rate of a hybrid approach (with the expanded signal set) that follows the criterion of re-training the system only when a disruption is missed. For the JET ILW campaigns, only 10 re-trains after disruption 42 have been necessary (instead of 159 as in figure 6). In figure 7, each re-training is represented by a circle. The numbers at the top of the figure show the number of disruptive discharges successfully recognized by each predictor before missing a disruption. A total number of 160 disruptions happened and 150 of them have been recognized (93.75%). The average rate of false alarms is 2.79%.

5. Summary and discussion

This paper discusses a specific method to develop disruption predictors from scratch. The term 'from scratch' means that the parameter space of the tokamak is completely unknown and, therefore, at the beginning there is no available information to train a predictor. This happens when a new machine starts its operation (let us think of JET with the ILW or ITER). New tokamaks are expected to start with discharges of low toroidal field and low plasma current where disruptions are not dangerous. In early campaigns, disruptions are expected to be a much larger fraction of the discharges. Therefore, the early campaigns will provide an excellent dataset to apply the adaptive predictors described here. As performance increases and disruptions become more dangerous, the adaptive learning method will provide an optimal method for minimizing disruptions and potential machine damage.

The analysis of the first three ILW campaigns in JET shows the good results of a hybrid methodology. The methodology to develop predictors from scratch can be summarized in two steps:

(1) First predictors are based on a balanced approach. A new predictor has to be generated after every disruptive discharge.
(2) After about 40 disruptions, predictors are based on an unbalanced approach and this unbalance is in favour of the disruptive discharges. A new predictor has to be generated after every missed alarm.

These conclusions come from the analysis of more than 1000 discharges (disruptive and non-disruptive) corresponding to JET ILW campaigns. But it is important to understand the implications of the reported results for the development of disruption predictors from scratch in future fusion devices (ITER and DEMO).

There are two different phases in the learning process. The first one is characterized by a fast learning rate. The objective is, on the one hand, to acquire a gross knowledge of the operational space (in relation to disruptions) as soon as possible. On the other hand, this knowledge has to be refined in the least possible time. According to figures 5 and 6, a few tens of disruptions are enough to have a good predictor system (high success rate and low false alarm rate). The second phase starts when a stable predictor has been obtained. In this context, 'stable' means to smoothly achieve both a minimum threshold of success rate and a maximum threshold of false alarms (figure 6). Of course, new disruptions are always possible and when they happen, a new predictor has to be developed in such a way that the new version includes not only all the previous knowledge about disruptions but also the information provided by the new disruption. In this sense, we think that the key point in a predictor is the training phase. Therefore, it is possible to obtain different results depending on several factors such as selection of discharges, methodology in the training phase, signal selection and so on. In any case, to evaluate the results, it is important not to forget that the models are obtained from scratch, that is, only the available discharges have been used without the possibility of choosing discharges from a large database. Taking this into account, the learning process is really fast.

It is also important to note that the tests carried out for the ITER-like wall campaigns involve a limited set of disruption classes and therefore the training might have gone quicker than if one had to handle more different types. This will be assessed in future campaigns.

Acknowledgments

This work was partially funded by the Spanish Ministry of Economy and Competitiveness under Project No ENE2012-38970-C04-01/03.

This work, supported by the European Communities under the contract of Association between EURATOM/CIEMAT, was carried out within the framework of the European Fusion Development Agreement. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

References

[1] Wesson J.A. *et al* 1989 Disruptions in JET *Nucl. Fusion* **29** 641–66

[2] Hender T.C. *et al* 2007 Progress in the ITER Physics Basis: chapter 3. MHD stability, operational limits and disruptions *Nucl. Fusion* **47** S128–202

[3] Schuller F.C. 1995 Disruptions in tokamaks *Plasma Phys. Control. Fusion* **37** A135–62

[4] De Vries P.C. *et al* 2009 Statistical analysis of disruptions in JET *Nucl. Fusion* **49** 055011

[5] De Vries P.C. *et al* 2012 The impact of the ITER-like wall at JET on disruptions *Plasma Phys. Control. Fusion* **54** 124032–40

[6] Yoshino R. 2005 Neural-net predictor for beta limit disruptions in JT-60U *Nucl. Fusion* **45** 1232–46

[7] Windsor C.G. *et al* 2005 A cross-tokamak neural network disruption predictor for the JET and ASDEX Upgrade tokamaks *Nucl. Fusion* **45** 337–50

[8] Cannas B. *et al* 2007 A prediction tool for real-time application in the disruption protection system at JET *Nucl. Fusion* **47** 1559–69

[9] Murari A. *et al* 2008 Prototype of an adaptive disruption predictor for JET based on fuzzy logic and regression trees *Nucl. Fusion* **48** 68–76

[10] Rattá G.A. *et al* 2010 An advanced disruption predictor for JET tested in a simulated real-time environment *Nucl. Fusion* **50** 025005

[11] Vega J. *et al* 2013 Results of the JET real-time disruption predictor in the ITER-like wall campaigns *Fusion Eng. Des.* at press doi:10.1016/j.fusengdes.2013.03.003

[12] De Vries P.C. *et al* 2011 Survey of disruption causes at JET *Nucl. Fusion* **51** 053018

[13] Japkowicz N. and Stephen S. 2002 The class imbalance problem: a systematic study *Intell. Data Anal.* **6** 429–49

[14] Weiss G.M. and Provost F. 2003 Learning when training data are costly: the effect of class distribution on tree induction *J. Artif. Intell. Res.* **19** 315–54

[15] López J.M. 2012 Implementation of the disruption predictor APODIS in JET real-time network using the MARTe framework *Proc. 18th IEEE-NPSS Real Time Conf.* doi:10.1109/RTC.2012.6418168

[16] Neto A.C. *et al* 2010 MARTe: a multiplatform real-time framework *IEEE Trans. Nucl. Sci.* **57** 479–86

[17] Rattá G.A. *et al* 2008 Feature extraction for improved disruption prediction analysis at JET *Rev. Sci. Instrum.* **79** 10F328

[18] Rattá G.A., Vega J., Murari A. and JET-EFDA Contributors 2012 Improved feature selection based on genetic algorithms for real time disruption prediction on JET *Fusion Eng. Des.* **87** 1670–8