IOP PUBLISHING and INTERNATIONAL ATOMIC ENERGY AGENCY NUCLEAR FUSION

Nucl. Fusion **50** (2010) 025005 (10pp) doi:10.1088/0029-5515/50/2/025005

An advanced disruption predictor for JET tested in a simulated real-time environment

G.A. Rattá[1], J. Vega[1], A. Murari[2], G. Vagliasindi[3], M.F. Johnson[4], P.C. de Vries[4] and JET EFDA Contributors[a]

JET-EFDA, Culham Science Centre, Abingdon, Oxon OX14 3DB, UK

[1] Asociación EURATOM/CIEMAT para Fusión. Avda. Complutense, 22. 28040 Madrid, Spain

[2] Associazione EURATOM-ENEA per la Fusione, Consorzio RFX, 4-35127 Padova, Italy

[3] Dipartimento di Ingegneria Elettrica Elettronica e dei Sistemi-Università degli Studi di Catania, 95125 Catania, Italy

[4] EURATOM/UKAEA Fusion Association, Culham Science Centre, Abingdon, Oxon OX14 3DB, UK

Received 5 June 2009, accepted for publication 17 November 2009
Published 15 January 2010
Online at stacks.iop.org/NF/50/025005

Abstract

Disruptions are sudden and unavoidable losses of confinement that may put at risk the integrity of a tokamak. However, the physical phenomena leading to disruptions are very complex and non-linear and therefore no satisfactory model has been devised so far either for their avoidance or their prediction. For this reason, machine learning techniques have been extensively pursued in the last years. In this paper a real-time predictor specifically developed for JET and based on support vector machines is presented. The main aim of the present investigation is to obtain high recognition rates in a real-time simulated environment. To this end the predictor has been tested on the time slices of entire discharges exactly as in real world operation.

Since the year 2000, the experiments at JET have been organized in campaigns named sequentially beginning with campaign C1. In this paper results from campaign C1 (year 2000) and up to C19 (year 2007) are reported. The predictor has been trained with data from JET's campaigns up to C7 with particular attention to reducing the number of missed alarms, which are less than 1%, for a test set of discharges from the same campaigns used for the training. The false alarms plus premature alarms are of the order of 6.4%, for a total success rate of more than 92%. The robustness of the predictor has been proven by testing it with a wide subset of shots of more recent campaigns (from C8 to C19) without any retraining. The success rate over the period between C8 and C14 is on average 88% and never falls below 82%, confirming the good generalization capabilities of the developed technique. After C14, significant modifications were implemented on JET and its diagnostics and consequently the success rates of the predictor between C15 and C19 decays to an average of 79%. Finally, the performance of the developed detection system has been compared with the predictions of the JET protection system (JPS). The new predictor clearly outperforms JPS up to about 180 ms before the disruptions.

PACS numbers: 52.25.−b, 29.85.Fj, 07.05.Mh

(Some figures in this article are in colour only in the electronic version)

1. Introduction and previous studies

Disruptions are major instabilities that can take place during a tokamak operation. They consist of sudden losses of confinement which cause the abrupt termination of the discharge. In addition to affecting the execution of the research programme, they can constitute a risk for the structural integrity of the machine [1]. Disruptions can be triggered by various instabilities which, on time scales even of the order of milliseconds, can force the plasma out of its safe operational limits with the resulting loss of energy and termination of the plasma current. The first phase of the disruption, the so-called thermal quench, can cause extremely high thermal loads on the plasma facing components and more in general on the first wall. As a consequence of the subsequent plasma abrupt current quench, large eddy currents can be induced in

[a] See the appendix of F. Romanelli *et al* 2008 *Proc. 22nd IAEA Fusion Energy Conf. (Geneva, Switzerland, 2008).*

the vacuum vessel and surrounding structures creating forces potentially capable of producing severe damage to the device. Besides, in typical elongated cross-section tokamaks, plasmas are inherently unstable against vertical displacements that can lead to vertical displacement disruptions (VDE). In a VDE the plasma moves into the wall and an ex-plasma current flow is developed. Those resulting in-vessel currents are commonly called 'halo currents'. Also, the production of relativistic (runaway) electrons during disruptions poses another threat to the integrity of the plasma facing components and their underlying substrate structures [2].

Up to now, the occurrence of disruptions has proven to be an unavoidable aspect of tokamak operation particularly in high performance configurations.

The physical characterization of disruptions for prediction and control is an extremely complex task. The amount of available signals in each pulse and the non-linear relationship between them have rendered impossible up to now the development of a physical model to reliably recognize and predict the occurrence of this hazardous plasma behaviour. Therefore in the last decade, various machine learning techniques, mainly artificial neural networks and support vector machines (SVMs), have been used as an alternative approach to disruption prediction [3–15]. These computational systems are capable of building general models, by 'learning' from the data in the so-called 'training process'. Once the systems have been trained, they can be applied to detect the specific behaviour they were designed to identify.

Unfortunately, and notwithstanding the considerable amount and quality of previous investigations, the obtained results have not been completely satisfactory. The most relevant deficiency, in most of the previous works, is that the complete evolution of the discharges is not analysed in its full length but only certain selected parts of the shots are considered. The models developed in this way are therefore useful to study the physics of the phenomenon but they cannot be applied in real-time environments, where the need of predicting disruptions extends to the whole evolution of the pulse. Moreover in some of the mentioned works, databases with less than 300 shots have been considered [3, 7, 8, 11–13]. Also, the determination of the predictor performance was usually limited to the period in which the actuators can undertake mitigation actions. Some disruptions cannot be identified by any algorithm, trained system or formula, even few milliseconds before the sudden loss of confinement takes place. These types of disruptions were normally discarded from the databases. In contrast, to assess the performance of the present predictor in a fully general way, these problematic discharges have been included both in the training and test sets for all the cases discussed in this paper.

One of the most advanced published studies performed so far at JET is the one by Cannas *et al* [8]. However, it was performed with a medium-size database (172 disruptive pulses and 102 non-disruptive pulses). Also, the premature alarms were not defined and the overall results are encouraging but still could be improved. In the cited work, the false alarms are minimized and the missed alarms are considered a secondary concern issue, prioritizing the continuity of the research programme over the protection of the device's integrity.

Another relevant work at JET that must by mentioned [9] is a neural network trained with 360 discharges which achieved 90% of successful detections with 5% of missed alarms. However, the testing shots were from the same period of experiments used for the training and therefore the robustness of the predictor for newer campaigns was not assessed.

Even if it refers to a different device (ASDEX-Upgrade) and therefore its applicability in JET has not been verified, it is worth mentioning the predictor developed by Pautasso [10]. It is based on the training of a neural network using eight plasma parameters and their time derivatives extracted from 99 disruptive discharges. Once trained, the system was tested firstly off-line with 500 shots attaining 85% of successful recognitions with 1% of missed alarms. Secondly the test was performed on-line during 128 pulses with 79% of correct identifications. In this last case, the fast types of disruptions were considered unrecognizable and discarded to calculate the statistics.

In an interesting piece of research [11], a neural network is trained with discharges of JET and the prediction system is tested at ASDEX-Upgrade with a success rate of a 67%. The converse test led to a 69% of correct recognitions at JET. The initial database for that work contains a wide range of shots, but only 175 are finally chosen.

Similar studies have been performed in other tokamaks. At DIII-D a neural network was trained to detect high β disruptions [12] attempting to set thresholds to trigger alarms. Several discharges were discarded due to the lack of a complete set of plasma parameters required by the system to perform the predictions, leaving a remaining 84 shots for the training and the testing. At ADITYA [13] a work aimed at predicting density limit disruptions, employed a database of 23 shots. This research only analyses disruptive shots and it cannot be applied in real time. Also with neural networks, two studies by Yoshino have been developed at JT-60U. In the first study [14] the system is trained in two steps. First, with 12 disruptive and six non-disruptive shots (step 1) and second (step 2), with the NN output data for 12 disruptive shots modified according to the output stability levels from the NN trained in step 1. The developed model was tested with 300 disruptive shots and 1008 non-disruptive shots attaining prediction rates higher than 80% at 50 ms prior to the disruption (however, it only analysed density limit disruptions, discarding the difficult types). The second study [15] uses a database of 525 discharges and has shown that training with non-disruptive shots and with an adequate adjustment of the step 2, obtains promising results in the identification of beta limit disruptions (the overall success rate reaches 76%).

In the present research the complete evolution of the considered discharges is followed from the beginning to the end to determine whether a disruption is forthcoming or not. The applied database is the biggest used so far at JET for studying real-time disruption recognition with learning systems. The database contains all types of disruptions in the considered campaigns. No type of disruption has been excluded and no bias has been introduced in the selection of the training and testing sets (they have been selected completely randomly). The final database consists of 2124 discharges (for which all the needed signals are available) and has been properly validated by experts.

Significant pre-processing of each signal, according to a procedure fully described in a previous work [4] and summarized in section 3, has been applied to every shot to extract the main features every 30 ms. This feature extraction procedure is performed to achieve high recognition rates with a minimum of missed alarms. This is motivated by the consideration that, in the perspective of ITER, particular attention will have to be devoted to disruption avoidance given the potential damage to the machine.

From a methodological point of view, two main innovative developments have been implemented to obtain the results described in this paper. First of all, a series of different predictors based on SVMs and trained during specific time intervals before the disruption have been deployed in parallel. The determination whether to trigger an alarm or not is obtained by a decision function (DF), again based on SVM, on the basis of the outputs of the individual predictors. This second layer of SVM, which is the second original aspect of the technique, is adaptive in the sense that it could be retrained automatically with the signals of each new discharge. However, the computational times could be improved. The training of the models of the first layer (with a Pentium 4 CPU, 3.2 GHz and using Matlab) requires about 1 h. The training of the second layer needs a computational time of ∼10 h. Finally, the model has been trained with shots of the campaigns up to C7 (discharge numbers from 42815 to 57346). It has been tested with discharges up to campaign C14 with very positive performances (practically no degradation in the success rates). After C14, JET has been subjected to structural modifications, in particular the replacement of the divertor and the bolometric diagnostic, and therefore the success rates from C15 to C19 are lower.

With regard to the structure of the paper, section 2 is devoted to a brief introduction to the SVMs and their conceptual basis. Section 3 describes the databases used and outlines the feature extraction method. Section 4 is focused on the training method, whereas the subject of section 5 is the development of the final DF, the step leading to or not leading to the triggering of an alarm. Section 6 reports the results obtained during the campaigns on which the predictor has been trained. Section 7 demonstrates the system generalization capability by testing the predictor with pulses of campaigns from C8 to C19. In section 8, a comparison between the developed predictor and JET protection system (JPS) is performed. A final discussion with the prospects for further investigations is the subject of the final section.

2. Introduction to SVMs and machine learning concepts

Artificial intelligence is a branch of science based on programming computers to solve complex problems in an intelligent way. Machine learning is the name given to a set of artificial intelligent techniques aimed at making computers 'learn'. By detecting significant patterns in the available data, a computational system based on these techniques can take decisions or make predictions about new data coming from the same physical object or source. In this sense, the system is able to acquire a generalization capability by 'learning' about the source generating the data.

Supervised methods are a category of machine learning techniques which can be applied to automatic classification. They usually work in two stages. In the first stage, called training, a set of known examples and the class they belong to are provided to the system. Once the training has been completed, a classifier is created. That classifier is used in the second stage, called testing. In this second phase, new examples are provided to the classifier, which determines the class each one belongs to.

For this work, the inputs are feature vectors corresponding to JET discharges and the possible classes are two: disruptive and non-disruptive. After the training, in the testing phase, the rate of successful classifications can be assessed in terms of percentages.

The supervised learning system utilized in this study is based on SVMs [16] because of their simplicity, good performance and generalization capability.

The following paragraphs are devoted to providing the basic concepts of SVM for classification problems.

Given the training objects x_i, each one of the class y_i such as

$$\{x_i, y_i\}; \qquad i = 1, \ldots, l; \qquad y_i \in \{+1, -1\}; \qquad x_i \in \Re^d. \quad (1)$$

A hyperplane that splits the learning objects into the classes y_i is calculated as

$$w \cdot x + b = 0,$$

where w is normal to the hyperplane. The distance from the hyperplane to the origin is $|b|/\|w\|$ and $\|w\|$ represents the Euclidean norm of w.

Defining d_+ and d_- as the distance from the separating hyperplane to the closest object (over and below the hyperplane, respectively), the margin will be the sum of d_+ and d_-.

Considering the following restrictions:

$$x_i \cdot w + b \geqslant +1 \qquad \text{for} \quad y_i = +1, \quad (2)$$

$$x_i \cdot w + b \leqslant -1 \qquad \text{for} \quad y_i = -1. \quad (3)$$

So

$$y_i(x_i \cdot w + b) - 1 \geqslant 0 \forall i. \quad (4)$$

The objects of (2) belong to the hyperplane $H_1 : x_i \cdot b = 1$ with a distance to the origin $|1-b|/\|w\|$. The objects of belong $H_2 : x_i \cdot b = -1$ with a distance to the origin $|-1-b|/\|w\|$.

Knowing that $d_+ = d_- = |1|/\|w\|$, the total margin is $|2|/\|w\|$.

H_1 and H_2 are parallel and no object is placed between them. It is possible to calculate both hyperplanes minimizing $\|w\|^2$ with the restrictions summarized in (4). For a bidimensional case, the solution is represented in figure 1(*a*).

Given a dimensional space it is possible to separate linearly quantity of objects equal to the dimension plus one. The proposed solution implements functions called kernels to transform the data to a higher dimensional space where a lineal separating hyperplane can be built. Representing a lineal function as

$$f(X) = \sum_{j=1}^{n} w_j \cdot w_j, \quad (5)$$

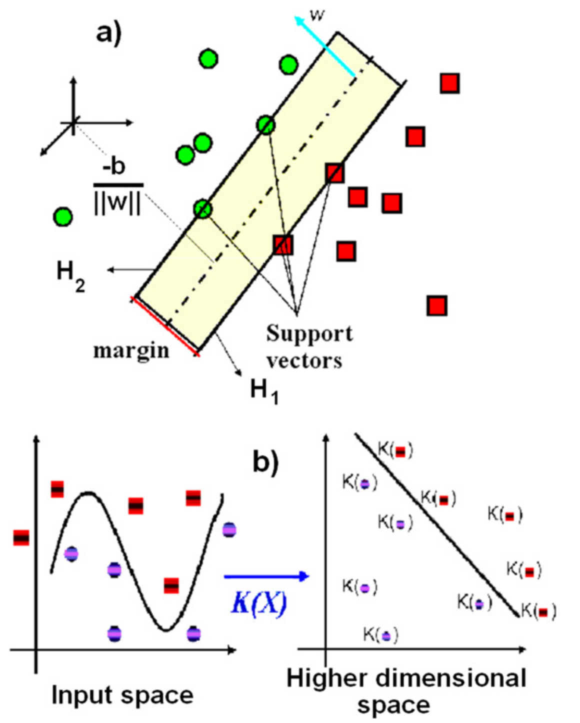

Figure 1. (*a*) Support vectors providing maximal separation margins in a linearly separable problem. (*b*) A non-linearly separable case solved with a linear separating hyperplane in a higher dimensional space.

and describing W as

$$W = \sum_{i=1}^{n} \alpha_i \phi(i), \tag{6}$$

where $\phi(i)$ is a mapping function. The dual form of the previous equations is

$$f(X) = \sum_{j=1}^{m} \left(\sum_{i=1}^{m} \alpha_i \phi(i) \right) x_j, \tag{7}$$

$$f(X) = \sum_{i=1}^{m} \alpha_i \sum_{i=1}^{m} \phi(x_i)\phi(x_j) = \sum_{i=1}^{m} \alpha_i K(x_i, x), \tag{8}$$

where K is the kernel function and $\alpha \in \Re^m$ represents expansion coefficients.

In this new space, the criteria for linear separation can be applied and the hyperplane can be created, see figure 1(*b*). The learnt solution is based only on those examples (support vectors) closest to the hyperplane.

Once this separating hyperplane has been calculated, it is used to determine the class of the new tested objects. The hyperplane splits the space into two and the objects are classified depending on which of the two regions of the space they belong to. Besides, extra information can be derived from this classification method: the distance between the objects and the hyperplane can be calculated and it can be considered an indicator of the classification confidence. If the distance between the new tested object and the hyperplane is large, this means that the classification is more reliable. If instead the distance is small, the new object is near the hyperplane and therefore closer to the boundary with the other class. A small distance, then, indicates a low reliability of classification.

Table 1. List of the databases and total number of discharges used for the analysis.

Shots (Campaigns)	Number of discharges
Disruptive-train (C7 and earlier ones)	**263**
Non-disruptive-train (C7 and earlier ones)	**175**
Disruptive-test (C7 and earlier ones)	**66**
Non-disruptive-test (C7 and earlier ones)	**44**
Disruptive-test (C1 to C19)	1245
Non-disruptive-test (C8 to C19)	331
Total disruptive (test and train)	**1574**
Total non-disruptive (test and train)	**550**
Total discharges (test and train)	2124

The implemented software [17, 18] used to obtain the results reported in this paper is included in public licensed environments for MATLAB® and C++.

3. The database

A large dataset extracted from the JET database (shots in the range from 42815 to 70722) has been utilized in this study. To the authors' knowledge, it contains the highest number of discharges ever considered for real-time disruption prediction studies using learning systems. A subset of the database has been used to build the predictor and to test it. In table 1 the text in bold indicates the portion of the database used for this purpose (training and testing datasets) with randomly selected shots of campaigns up to C7. Once the system has been trained and tested (sections 4, 5 and 6), its robustness has been proved with a wide number of shots from C1 (year 2000) to C19 (year 2007). The quality of the predictions is finally compared with the ones provided by the JPS (shots from C1 to C19). The number of disruptive and non-disruptive pulses used in the testing stages is detailed in the rest of table 1.

At 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 prediction. Thirteen signals to study disruptions have been chosen for training and testing the classifier (see table 2). The results shown in a previous work [4] proved that with this selection of signals, good results can be obtained. Besides, the majority or all of this set (that includes some time derivatives of the signals which can improve the recognition of the phenomenon) have also been used in previous research on the subject [6, 8].

Because each measurement has been acquired by a different diagnostic, their sampling rates are not necessarily the same. Therefore a simple interpolation algorithm has been applied to standardize the sampling rate to 1 kHz. It is necessary to highlight that each of the 13 signals is required and consequently the shots, for which any of these measurements are not available, have not been considered. The percentage of discarded shots has been about 8.5% mainly due to the lack of reliable total radiated power estimates. The developed

Table 2. List of the signals analysed in each shot.

Signal name		Units
(1)	Plasma current	A
(2)	Poloidal beta	
(3)	Poloidal beta time derivative	s^{-1}
(4)	Mode lock amplitude	T
(5)	Safety factor at 95% of minor radius	
(6)	Safety factor at 95% of minor radius time derivative	s^{-1}
(7)	Total input power	W
(8)	Plasma internal inductance	
(9)	Plasma internal inductance time derivative	s^{-1}
(10)	Plasma vertical centroid position	m
(11)	Plasma density	m^{-3}
(12)	Stored diamagnetic energy time derivative	W
(13)	Net power (total input power *minus* total radiated power)	W

prediction system will trigger an alarm to halt the discharge if a signal is lost during real-time operation.

Another potential issue is the extreme amplitude difference between some signals, in many cases of several orders of magnitude. This aspect is a complicating factor for any classification system. The system could automatically assign higher weights to certain signals because of their absolute values and not because of their relevance for the prediction. Consequently, a standard normalization formula has been applied:

$$\text{Normalized Signal} = \frac{\text{Signal-Min}}{\text{Max-Min}},$$

where Min and Max, respectively, represent the minimum and maximum values of each signal in the training set. This normalization rescales the signal values to the interval between 0 and 1 preserving the information about their relative magnitudes.

To demonstrate that a predictor provides sufficient and reliable performances, it is necessary to train and test it using a considerable number of discharges from a large database of stored signals. To properly assess the potential of the predictor for control applications, it has been designed to use the data sequentially as if they became progressively available during an actual experiment. For that, 30 ms time windows of each plasma parameter are considered. Each time window is processed in order to condense the disruptive related information in feature vectors. Each feature vector contains 13 values (one per signal). For each signal, the feature extraction procedure consists of the calculation of the fast Fourier transform. The positive part of the spectrum is retained, discarding the first component (off-set). The standard deviation of the retained spectra is calculated, attaining one value. Finally, the 13 values are concatenated to attain the feature vector.

This feature extraction procedure has been chosen since it provides the best performance as reported in detail in [4].

Using these feature vectors, the ultimate action of taking a decision (to trigger or not to trigger an alarm) is performed as if they were being acquired and processed in real time.

4. Training process

Various classifiers based on the SVM have been trained to identify forthcoming disruptions. The training has been performed by providing the learning systems with two classes of inputs: features belonging to disruptive discharges and features of non-disruptive discharges.

Particular attention has been devoted to avoiding any form of bias in the selection of the training set. A balanced number of disruptive and non-disruptive shots (263 and 175, respectively) of the database have been chosen randomly for the training process.

The notation used to identify the models can be summarized as follows:

M(i) is the model trained for a specific time interval i before the disruption.
where $i = 1, 2, 3, \ldots, 8$
M(1) corresponds to the period [−60,−30] ms before the disruption.
The minimum time to perform mitigation actions at JET is 30 ms [19]. The detection system must be trained to recognize disruptions at least 30 ms before their occurrence. Consequently, the closest period to the disruption [−30, 0] has been discarded for the training.
The others M(i) refer to the intervals $[-30(i+1), -30i]$ ms before the disruption.
The training inputs, for each model have been the following.
As *disruptive features*: time windows $[-30(i+1), -30i]$ ms before the disruption for all shots in the disruptive training dataset.
As *non-disruptive features*:

- All the time windows from the beginning of the shot to 1 s before the time of the disruption for all the pulses in the disruptive training dataset. Before those times even disruptive shots do not show clearly abnormal behaviours and they could be considered non-disruptive, as it has been proven in [4]. Besides, the alarms triggered before 1 s prior the disruptions can be considered as 'prematures', interrupting the continuity of the operation [7].
- All the time windows of the non-disruptive discharges available in the non-disruptive training dataset.

To optimize the success rates, different combinations of the trained classifiers have been considered. The choice to implement more than one classifier is due to the need to take into account the temporal evolution of the discharge. To this end, sequences of consecutive classifiers, optimized for various time intervals of 30 ms before the disruptions, have proved to achieve higher performance than a single model trained over a longer period. The classifiers are meant to operate in parallel on consecutive time windows as shown in figure 2. The number of predictions provided by this concatenation of n models (with $n = 3$ in the example of figure 2) is n.

At this point the pending issue is the determination of the optimal number of classifiers to be used in parallel. Because of the novelty of this approach, this number n was unknown and consequently different combinations have been tested. To this end eight classifiers have been developed (models trained

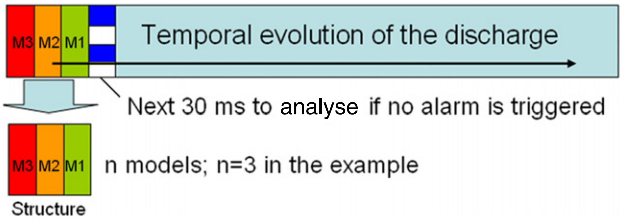

Figure 2. Scheme of the real-time analysis of a discharge using a series of classifiers analysing in parallel consecutive time windows.

with earlier time intervals have very poor performance due to the fact that not enough information is contained in the signals too far away from the disruption). With these models, seven sequences that include consecutive classifiers have been created concatenating them as follows:

Sequence 1: M2, M1.
Sequence 2: M3, M2, M1.
Sequence 3: M4, M3, M2, M1.
Sequence 4: M5, M4, M3, M2, M1.
Sequence 5: M6, M5, M4, M3, M2, M1.
Sequence 6: M7, M6, M5, M4, M3, M2, M1.
Sequence 7: M8, M7, M6, M5, M4, M3, M2, M1.

For the sake of clarity, the whole training procedure, as implemented for sequence two, is summarized in the following pseudocode:

WHILE (end condition, detailed in section 6, is not satisfied)

FOR (the whole training database)

1. The first classifier of the sequence (M3) analyses the first 30 ms of each shot, the second classifier, the seconds 30 ms and the third the third 30 ms (see again figure 2).
2. The three outputs of the sequence (one per classifier included in the sequence) are analysed with a DF, consisting of another SVM classifier (described in detailed in section 5). This DF determines whether the combination of outputs provided by the sequence indicates a forthcoming disruption or not.

IF (a disruptive behaviour is recognized)
An alarm is immediately triggered and the relevant data are stored for future analysis in an archive similar to table 3 (figure 3).

ELSE (no disruptive behaviour is recognized)
No alarm is triggered, no data are stored and the next 30 ms of discharge are analysed. Then, model 3 will analyse the time windows previously analysed by model 2, model 2 will analyse the time windows previously analysed by model 1 and model 1 will analyse the next incoming 30 ms of the shot, (returning to 2).
END IF

END FOR

END WHILE

5. The decision function

The n output values of the classifiers do not always coincide, i.e. some models predict an incoming disruption and others do not. Then, a DF has to be implemented to determine automatically whether an alarm has to be triggered or not. The development of this function is crucial to attain the best possible recognition rates for the global predictor. The accurate determination of what relationship of the n results provides the best performances can be performed automatically by a classification system. Consequently, the SVM was again applied for this purpose.

In the adopted procedure, the outputs provided by the individual classifiers described in section 4 are used to train the classification system that implements the actual DF. This final DF is obtained through an iterative process. First, a set of conditions has been empirically formulated to determine the initial decision rule (IDR), which allows performing the first discrimination of the discharges in disruptive and non-disruptive (see later). The IDR is just applied in the first step. Then an iterative procedure has been implemented to converge on an optimized DF. This procedure performs the analysis of the shots to obtain refined results to train the next DF. The process continues till an ending condition is activated.

The IDR has been formulated empirically. A considerable time consuming analysis has been performed to interpret the results of the n classifiers and to set the empirical conditions that decide the triggering of an alarm. It is also worth mentioning that there is a different IDR for each sequence of classifiers. For sequence 2 (M3, M2, M1) the IDR is described in the following, where the Vi are the distances of the Mi outputs to the separating hyperplanes.

If ($V3 > 0$ & $V2 > -0.4$ & $V1 > -0.8$)
The alarm must be triggered.

Instead, for sequence 1 (M2, M1) the conditions are
If ($V1 + V2 > 0$)
OR
If ($V2 > 0.3$)
The alarm must be triggered.

Obviously, this combination of empirical rules is not the most accurate procedure to achieve the highest recognition rates (applying the IDR to sequence 2, a success rate of 80% is obtained).

At the end of the first iteration the stored results are employed to train the next DF (figure 3). The criterion consists of saving the relevant data every time an alarm is triggered. The results fill a table that contains a number of rows equal to the number of analysed discharges.

In table 3 the six possible types of results that can occur in practice are summarized. The first column represents an index, the second the shot number, the third the time when the system has detected a disruptive behaviour, the fourth the real disruption time (0 if the pulse is non-disruptive), the fifth is the difference between columns three and four and the last three columns report the values provided by each model of the sequence.

Table 3. Examples of the six possible cases of correct and incorrect classifications that can occur in practice.

No	Shot	Detection time (s)	Disruption time (s)	Margin (ms)	Output of M3	Output of M2	Output of M1
1	56658	23.911	24.021	110	**−0.910**	**0.448**	**0.0852**
2(a)	54827	0	13.866	0	**0**	**0**	**0**
2(b)	55253	23.101	22.979	−122	**−1.800**	**−1.391**	**−1.154**
3	53740	5.881	10.22	4339	**−0.896**	**−1**	**0.445**
4	56782	0	0	0	**0**	**0**	**0**
5	52641	0.916	0	−916	**−1.176**	**−1.202**	**−1.008**

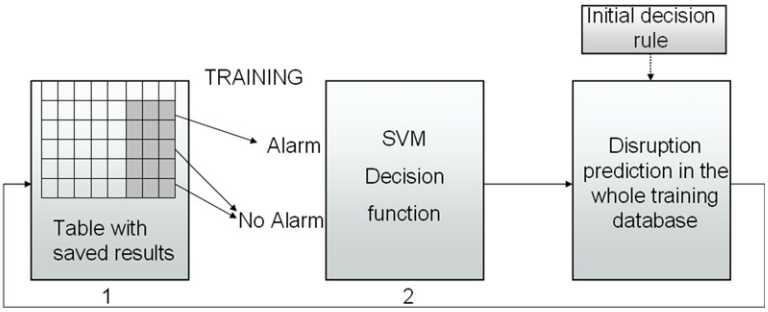

Figure 3. General DF training scheme. 1. The relevant data are stored. 2. A new DF that replaces the previous one is trained.

These results are just examples obtained from one iteration of the optimization process that leads to the development of the final DF. The rows represent:

(1) An incoming disruption correctly recognized 110 ms before it occurs.
(2) Two possible cases of missed alarms:
  (a) A disruptive behaviour not recognized in a disruptive shot.
  (b) A disruption that is recognized after the occurrence of the disruption.
(3) A premature alarm: a disruptive behaviour is detected in a disruptive shot too much in advance (more than 1 s). In the example 4339 ms before the disruption occurs.
(4) A non-disruptive pulse where, correctly, no disruptive behaviour has been recognized.
(5) A false alarm: a non-disruptive discharge where, incorrectly, a disruptive behaviour has been recognized.

The outputs of each classifier (the three last columns of the example in table 3) of the sequence are given to the SVM learning system to create the DF. The outputs for training the DF can be divided into two groups or classes. *The first class* (alarm activation) contains all the models output values of case 1 (table 3), i.e. all the cases where the system has correctly determined an incoming disruption. With these examples the system learns when an alarm must be triggered. *The second class* (NO alarm activation) contains all the model output values of cases 3 (premature alarms) and 5 (false alarms) of table 3. With these examples the system is trained to neither activate an alarm too early in disruptive shots nor in non-disruptive shots.

After each iteration a new DF is trained. Then this new DF replaces the previous one. Also a new table, similar to table 3, is obtained and added to the previous ones. These new data are used for training the next DF. In this way, at every new iteration, more data are entered to refine the previous DF.

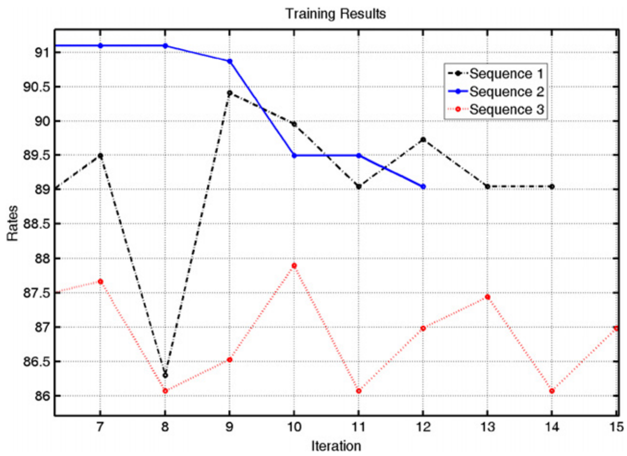

Figure 4. Training results of three sequences. The training procedure is stopped after five iterations without any improvement.

Its optimization continues till the ending condition (the success rates obtained with the DF do not improve after five iterations of the complete training process) is activated (see figure 4).

6. Testing

The testing phase was performed with a different subset of discharges belonging to the same period that the predictor was trained with. As in the training case, the shots have been divided in time slices of 30 ms from the start of the shot. All seven sequences of section 4 have been tested. For each of these sequences, the most performing DF is utilized to identify disruptive behaviours in the testing stage. Finally, the obtained results are employed to compute statistics for each sequence.

During the training phase a clear tendency has been noticed: sequences that concatenate more than three models do not provide the best classification. The best performance is not

Table 4. Overall test results.

	MA	FA	PA	TE	SR	AVG
Predictor 1 [Sequence 1, DF1]	0	5.4545	5.4545	10.909	89.091	128.01
Predictor 2 [Sequence 2, DF2]	**0.909**	**4.5455**	**1.8182**	**7.2727**	**92.727**	**146.23**
Predictor 3 [Sequence 3, DF3]	0	4.5455	5.4545	10	90	132.23
Predictor 4 [Sequence 4, DF4]	0	5.4545	9.0909	14.545	85.455	128.84
Predictor 5 [Sequence 5, DF5]	0	4.5455	7.2727	11.818	88.182	136.45
Predictor 6 [Sequence 6, DF6]	0	5.4545	5.4545	10.909	89.091	136.23
Predictor 7 [Sequence 7, DF7]	0	5.4545	6.3636	11.818	88.182	121.01

achieved using shorter sequences either. The higher success rates are obtained with sequence 2 which seems to strike the proper trade-off between complexity and simplicity.

However, the performances of the classifiers can only be evaluated reliably in the testing phase. When the predictor new cases to classify are given to the predictor, the results show again that the best sequence remains as number 2 (see table 4). Sequence 2, even allowing a small percentage of missed alarms (0.9%), commits a considerably lower rate of premature plus false alarms in all cases. It demonstrates that the sequence prioritizes the attainment of the highest overall results. Its superior performances can also be appreciated in terms of the average recognition times (AVG). In the table, from left to right, each column represents the classifiers and the rates of missed alarms (MA), false alarms (FA), total errors (TE) and success rates (SR) expressed in percentages. AVG represents the mean time, expressed in ms, between the triggering of the alarms and the occurrence of the actual disruptions for the whole testing set. The typical times at JET to perform mitigation actions are between 30 ms and 200 ms [19].

7. Testing the system with shots of more recent campaigns

In section 2 it was mentioned that the SVM has generalization capabilities. To confirm the potential of the developed predictor in this respect, it has been tested with pulses of the campaigns C8–C19. It is worth mentioning that after C14, significant changes have been implemented on the device, including the MKII GB LBSRP (Mark II Gas Box Load Bearing Septum RePlacement) installation and the substitution of the previous bolometer system with a new one, which provides signals different from the ones the predictor was initially trained with. Also, the error field correction coils (EFCCs) [20], that can affect the mode lock signal, have been installed to change the error field level and to measure the beta limit.

The final results have been summarized in figure 5. The light shadowed rectangle indicates the campaigns in which the model has been previously trained and validated. The non-shadowed middle part of the figure displays the percentages that have been computed over 376 shots (50% of them are disruptives). The variation in success rate is below 11.6% (between 94.12% and 82.35%). In that interval (C8 to C14) the minimum success rate has been obtained in C11, the Trace Tritium Experiment Campaign at JET. Because of the restrictions in the use of tritium, stringent operation rules were followed in C11. Due to the operational constraints, considerably fewer disruptions were made. Typical disruptions that usually take place during other campaigns have not happened. Many of the disruptions during this campaign were fast and difficult to classify.

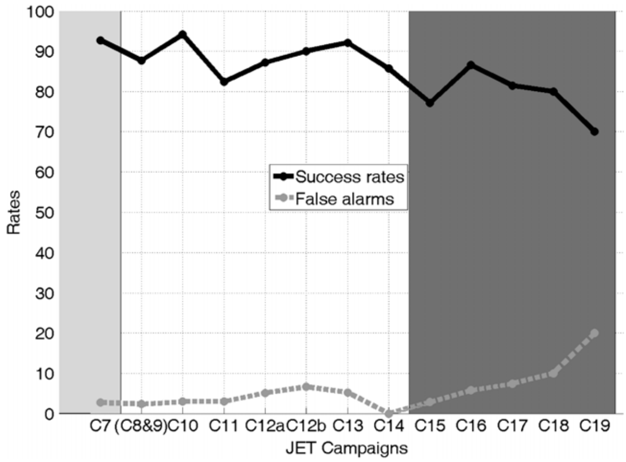

Figure 5. Success and false alarms rates of the previously trained model in posterior campaigns. Three different periods have been independently analysed. First (left), the training and validation period that includes campaign C7 and earlier ones (light shadowed). Second, the testing period between campaigns C8 to C14 (middle, not shadowed), third (right), after structural changes of the device, the period between C15 to C19 (dark shadowed part of the graph).

Finally, the dark shadowed portion of figure 5 illustrates the results of campaigns C15–C19 (246 discharges, half of them being disruptives). The cause of many of the false alarms after C14 can easily be found, such as the use of EFCCs affecting the mode lock signal and thus confusing the SVM detection system. If action is taken to prevent such false alarms the performance would look considerably better. But we have decided to leave these to give a realistic trend of the performance for a working device to which new components are often added. Besides, from C15 till the beginning of C16 the machine was subjected to an unusual increment of operational problems. UFOs, dirty plasmas and even leaks in the vacuum vessel interrupted the research programme. The worst rates have been attained in C19 mainly due to the noticeable increment of false alarms. About 47% of those alarms are caused by anomalies detected by the predictor. In these experiments the EFCCs were used and probably affected the locked mode signal. ∼44% of the other false alarms are triggered by the detection of large radiation spikes, that can be measured by the new bolometer diagnostic but not by the previous one the system was trained with. Therefore, the SVM detection system wrongly identified these sudden and unlearned increments of the radiation as disruption precursors.

It is worth emphasizing that the stability of the performance over different campaigns, mainly from C8 to C14, not used in the training of the predictor, is indeed a positive and new result. After C14 the results are still encouraging because the causes of the increment in the false alarms have been recognized and therefore they can be corrected in a future predictor.

8. Comparison with the system available in JET

The performances of the predictor have been compared with the results of the JPS [21], which is the system used in real time on JET for the generation of alarms for incoming disruptions. Most of the alarms triggered by the JPS are due to large mode lock signals [14]. This system has been implemented in JET for avoidance strategies in all discharges and therefore is the main reference tool.

For a proper comparison of both systems, some issues must be taken into account. The JPS intervenes during the execution of an experiment every time it considers that a disruptive activity has been detected [19]. Due to this direct intervention on the plasma evolution, it is impossible to calculate statistics of false alarms and therefore only disruptive pulses have been taken into account for the comparison. Also for comparison purposes, the premature alarms have not been measured as previously but in the same manner as they are computed by the JPS. The rates are calculated in relationship to the 'warning times'. The warning times (the difference between the disruption time and the alarm time) are relevant because they represent the temporal margin the actuator has to perform avoidance or mitigation actions during the execution of a pulse. Each point on the curves plotted in figure 6 is the accumulated percentage of recognized disruptions with a warning time equal or higher than that specified in the corresponding x-axis.

Using the JET disruptive database, it has been possible to test the SVM system over all the disruptive shots included in the period between the beginning of campaign C8 till the end of JET's campaign C19 (483 disruptions, 347 of them unintentional).

In a real-time scenario, the detection of unintentional disruptions (the ones which occur during normal operation) requires a higher level of attention than the recognition of intentional disruptions (the ones explicitly triggered during sessions devoted to studying their physics). To check the results obtained for each case, separated and combined statistics over unintentional disruptions and intentional disruptions have been calculated. The results depicted in figure 6 show that notable higher recognition percentages are achieved with the SVM predictor for both unintentional and intentional disruptions. The SVM system provides significantly higher rates than the JPS in the time interval up to ∼180 ms before the disruption. In the interval between ∼180 ms and ∼0.9 s the JPS shows a better accuracy. This behaviour is coherent with the training of the system, based on examples covering the interval up to 200 ms before the disruptions. On the other hand some of the JPS's missed alarms may have been due to bad acquisitions of the mode lock signal, one of the main reference measurements analysed by the JPS [19]. The JPS system statistics have been computed

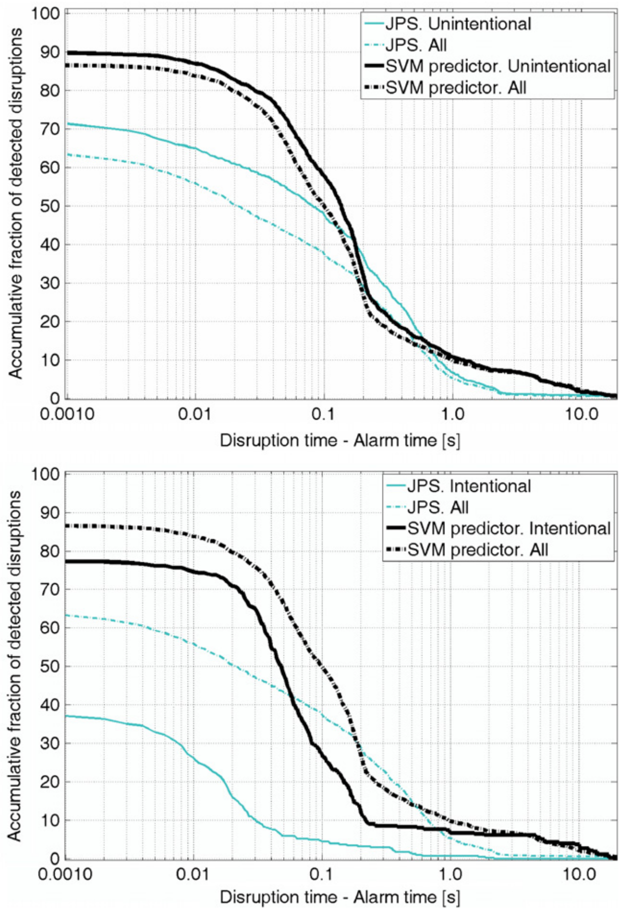

Figure 6. Warning times (disruption time minus alarm time) for all the disruptive pulses from campaign C1 till campaign C19. Top figure: the cumulative percentages of unintentional and all detected disruptions for the SVM method are compared with the JPS results. Bottom figure: the cumulative percentages of intentional and all detected disruptions for the SVM method are compared with the JPS results.

over all the discharges from C1 to C19, even with possible corrupted data, whereas the SVM predictor discarded 8.5% of the shots due to the lack of some required signals of the analysed pulses.

9. Summary and discussion

To summarize, several sequences of learning systems that analyse the complete evolution of each pulse have been developed to achieve the highest possible disruption prediction rates. The final predictor has been trained and tested on all types of disruptions from the campaigns up to C7. Subsequently it has been tested over a wide range of shots from more recent campaigns not used for the training. Finally the performance of the SVM predictor has been compared with the JPS.

In the testing stage, that included campaign C7 and earlier campaigns, with the best performing classifier, the overall success rate is 92.73%, with a 0.91% of missed alarms. This percentage of missed alarms is the result of one error over 110 shots (66 disruptives and 44 non-disruptives). The same rate computed only over the disruptive experiments for the same campaigns would be 1.52% (one missed alarm over 66 disruptive shots). The missed alarm was triggered in a discharge (shot #44116) with an abrupt termination due to a mistake in the plasma current control. In general, the missed alarms have been minimized and the sum of false and premature alarms was reduced to less than 6.4%.

The robustness of the predictor has been proven by testing it with a significant number of pulses belonging to more recent campaigns than the ones it was trained with. The relevance of this aspect is significant because the lack of generalization capability across campaigns was one of the main weaknesses of previous predictors. The results show clearly that the high performances remain fairly constant until a major modification of the device has been implemented. After that, the success is still high but the percentage of errors increases to 9%.

Finally, the developed method has been compared with the JPS used at JET for many years. Considerably higher performances have been attained in the interval up to ∼150 ms before the disruption. These are especially relevant results due to the fact that the typical time required by the actuators to perform effective mitigation actions at JET is about 30 ms.

To conclude, it is worth mentioning that in this work the highest priority has been assigned to reducing the number of missed alarms in order to preserve the integrity of the device. A different balance between these requirements could be reconsidered and the classifier optimized for different operational constraints.

With regard to future developments, a second stage of the predictor can be trained to determine the type of the disruption once an alarm has been triggered. This future research will require a complete revision of the feature extraction procedure (the characteristics to distinguish the type of a disruption may not necessarily be the same ones employed for its detection).

In the perspective of ITER, the next step should be to look for device-independent parameters (adimensional variables) in order to scale the results to any machine.

Acknowledgments

This work was partially funded by the Spanish Ministry of Science and Innovation under the Project No ENE2008-02894/FTN.

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] Schuller F.C. 1995 Disruption in tokamaks *Plasma Phys. Control. Fusion* **37** A135–62
[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] Cannas B. *et al* 2006 Disruption prediction at ASDEX Upgrade using neural Networks *33rd EPS Conf. on Plasma Physics (Rome, Italy, 19–23 June 2006)* ed F. De Marco and G. Flad vol 30 I (ECA) (Geneva: European Physical Society) P-2.143 http://epsppd.epfl.ch/Roma/pdf/P2_143.pdf
[4] Rattá G.A., Vega J., Murari A. and Johnson M. 2008 Feature extraction for improved disruption prediction analysis at JET *Rev. Sci. Instrum.* **79** 10F328
[5] Cannas B. *et al* 2004 Disruptions forecasting at JET using neural networks *Nucl. Fusion* **44** 68–76
[6] Murari A. *et al* 2008 Prototype of an adaptive disruption predictor for JET based on fuzzy logic and regression trees *Nucl. Fusion* **48** 035010
[7] Cannas B., Delogu R.S., Fanni A., Sonato P. and Zedda M.K. 2007 Support vector machines for disruption prediction and novelty detection at JET *Fusion Eng. Des.* **82** 1124–30
[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] Milani F. 1998 Disruption prediction in JET *PhD Thesis* University of Aston, Birmingham
[10] Pautasso G. *et al* 2002 On-line prediction and mitigation of disruptions in Asdex Upgrade *Nucl. Fusion* **42** 100–8
[11] Windsor C.G. *et al* 2005 *Nucl. Fusion* **45** 337
[12] Wroblewski D., Jahns G.L. and Leuer J.A. 1997 *Nucl. Fusion* **37** 725
[13] Sengupta A. and Ranjan P. 2001 *Nucl. Fusion* **41** 487
[14] Yoshino R. 2003 *Nucl. Fusion* **43** 1771
[15] Yoshino R. 2005 *Nucl. Fusion* **45** 1232
[16] Cortes C. and Vapnik V. 1995 Support-vector networks *Mach. Learn.* **20** 273–97
[17] The Spider - A machine learning in Matlab. http://www.kyb.tuebingen.mpg.de/bs/people/spider/main.html Max-Planck Institute for biological Cybernetics, Tuebingen, Germany
[18] Briggs T. 2005 MATLAB/MEX Interface to SVMlight. www document, http://www.ship.edu/∼thb/mexsvm/
[19] de Vries P.C., Johnson M.F. and Segui I. 2009 Statistical analysis of disruptions in JET *Nucl. Fusion* **49** 055011
[20] Barlow I. *et al* 2001 The error field correction coils on the JET machine *Fusion Eng. Des.* **58–59** 189–93
[21] Santagiustina A. *et al* 1995 Studies of tearing mode control in JET *Proc. 22nd EPS Conf. on Controlled Fusion and Plasma Physics (Bournemouth, 3–7 July 1995)* (European Conference Abstracts) 19c IV p 461 (EPS, Geneva)