A distributed Self-adaptive Nonparametric Change-Detection Test for Sensor/Actuator Networks

Cesare Alippi, Giacomo Boracchi, Manuel Roveri

Dipartimento di Elettronica e Informazione
Politecnico di Milano, Milano, Italy
{cesare.alippi, giacomo.boracchi, manuel.roveri}@polimi.it

Abstract. The prompt detection of faults and, more in general, changes in stationarity in networked systems such as sensor/actuator networks is a key issue to guarantee robustness and adaptability in applications working in real-life environments. Traditional change-detection methods aiming at assessing the stationary of a data generating process would require a centralized availability of all observations, solution clearly unacceptable when large scale networks are considered and data have local interest. Differently, distributed solutions based on decentralized change-detection tests exploiting information at the unit and cluster level would be a solution. This work suggests a novel distributed change-detection test which operates at two-levels: the first, running on the unit, is particularly reactive in detecting small changes in the process generating the data, whereas the second exploits distributed information at the cluster-level to reduce false positives. Results can be immediately integrated in the machine learning community where adaptive solutions are envisaged to address changes in stationarity of the considered application. A large experimental campaign shows the effectiveness of the approach both on synthetic and real data applications.

Keywords: Change detection test, sensor/actuator networks, fault detection.

1 Introduction

Networked embedded systems and sensor/actuator networks designed to work in real-life environments, e.g., water distribution systems, intelligent buildings, critical infrastructure networks, are subject to faults and ageing effects which generally induce a change in the statistical properties of the data coming from the field. Anticipating the detection of a change is a key issue to support intervention and avoid possible fault effects that would induce critical, when not catastrophic, consequences.

Change detection through quantitative observations of process variables is a hot research topic largely addressed in the literature. Among different strategies, refer to [1] for a comprehensive review, of particular interest are Change-Detection Tests (CDTs) [2][3], i.e., statistical techniques aiming at assessing satisfaction of the stationarity hypothesis for the data generating process. These tests generally assume

This research has been funded by the European Commission's 7th Framework Program, under grant Agreement INSFO-ICT-270428 (iSense).

that observations are all available in a centralized fashion at the decision making unit. This situation, even if acceptable in some cases, is not valid in general since it requires a large data exchange of no interest to the user with a communication cost. The problem amplifies in those remote parts of the network where energy is an issue despite the presence for energy harvesters, i.e., based on photovoltaic cells. To overcome such aspects, intelligence, here to be intended as the change-detection ability and solution adaptation, must be distributed among units which cooperate at different abstraction levels to build up a decision.

In the literature, distributed CDTs, can be grouped as distributed detection tests with quantized observations and local decisions. In both cases, all sensing units communicate with a centralized decision making unit (called data-aggregation center), whose goal is to aggregate data to provide the final decision.

Tests following the former approach (e.g., [4]) rely on the quantization of observations at the sensing units followed by a centralized analysis of such data at the central station. The use of quantized observations, which aims at controlling the power consumption in communication by reducing the bits to be transmitted does not generally provide the expected advantage in real-world applications since the bit savings is a small percentage of the packet size.

Differently, distributed tests following the latter approach (e.g., [5][6]) rely on independent local CDTs at the unit level whose decisions are transmitted to the unit for aggregation and decision. This philosophy fits well with distributed sensor/actuator networks. While tests at the sensing-unit level can be any traditional (centralized) CDT acting on available data, the key issue is how to effectively aggregate local decisions to achieve a final decision with low false positives and negatives. The works present in the literature are straightforward and generally propose to wait for a fixed number k of detections from the sensing units before raising a global variation in the data generating process. Assuming N sensing units in the network, traditional solutions set k equals to 1 (i.e., the distributed CDT detects a variation when at least one sensing unit detects a change) or N (i.e., the distributed CDT detects a variation when all the sensing units detect a change). Other values of k could be considered as well; e.g., [7] suggests $k=N/2$. Obviously, the value of k greatly influences the detection performance of the distributed CDT. In fact, low values of k guarantee low detection delays at the expenses of higher false positives. On the contrary, as k increases, detection delays increase but false positives decrease.

This work proposes a novel distributed change-detection test where each sensing unit performs a CDT and, once a change is detected, a second CDT algorithm is activated at the data-aggregation center receiving data from the network's cluster. If the change is confirmed, the CDT signals a global change in the network, otherwise the local detection is considered to be a false positive and the sensing units providing the wrong assessment are retrained to improve subsequent detection accuracy. We comment that the proposed distributed CDT improves over the distributed tests present in the literature by substituting the voting mechanism on k with a theoretically sound mechanism exploiting correlations among acquired observations. Furthermore, the two-levels distributed architecture allows us for being very reactive and prompt in detections while keeping under control occurrences of false positives.

The paper is organized as follows: Section II introduces the problem statement, while the suggested distributed CDT is presented in Section III. Experiments on synthetic and real applications data are shown in Section IV.

2 Problem Statement

Consider a sensor/actuator network composed of N sensing units that observe the same physical phenomenon, and a data-aggregation center that is connected with all units. The i^{th} sensing unit acquires over time data from a process $X_i: N \rightarrow R$, which generates independent and identically distributed (i.i.d.) observations drawn from an unknown probability density function (pdf). Under specific assumptions, the case of dependent observations w.r.t time could be brought back to this framework by means of suitable transformations (e.g., the i.i.d. assumption may concern the parameters of a model describing the data under suitable hypotheses [9] or innovation [3]). We outline that we do not require independence or the same distribution among observations coming from different sensing units. Let $O_{i,\mathrm{T}}=\{X_i(t),\ t=1,\ldots,\mathrm{T}\}$ be the sequence of observations acquired by the i^{th} unit up to time T, and assume that (at least) the first $\mathrm{T}_0<\mathrm{T}$ observations acquired by all sensing units have been generated by the process in a stationary state. Thus, O_{i,T_0} represents the training set of the i^{th} unit, i.e., a set of observations used to configure the parameters of the test. The proposed approach is nonparametric and, therefore, the pdfs of the data acquired in all the sensing units are unknown, both before and after the change. Moreover, after the change, the pdfs description can be time independent (e.g., abrupt changes) or evolve with time (e.g., drifts). We assume that the X_is under monitoring change their statistical properties at time instant $t=\mathrm{T}^*$ and that all sensing units can, potentially, observe the change. However, no assumptions are made about the effect of this change on each unit, as the magnitude and the profile of the change may vary from unit to unit (e.g., it may even affect only few nodes).

3 Distributed Change-Detection Test

The proposed distributed CDT relies on N units that independently monitor the process by means of unit-level CDTs, and by a data-aggregation center, which validates local detections by analyzing information sent from each node. In particular, the designed solution follows the approach delineated in [8] and combines the ICI-based CDT [10] at the unit-level, with a hypothesis test based on the Hotelling T-square statistic [11] at the data-aggregation center.

The proposed solution is outlined in Algorithm 1: during the training phase the ICI-based CDTs in execution on sensing units are configured using their respective training sequences (line 1). These tests monitor the process by extracting ad-hoc features from the observations: features extracted from O_{i,T_0} are then sent to the data-aggregation center (line 2), as they characterize how the i^{th} unit perceives the stationary process. During the operational life (line 4), for each new observation, each

Algorithm 1: Distributed Change-Detection System

1. *Each unit:*configure the ICI-based CDT using $\{O_{i,T_0}, 1 \le i \le N\}$;
2. *Each unit:* send feature extracted from O_{i,T_0} to the aggregation center.
3. **while**(units acquire new observations at time T){
4. *Each unit* runs the ICI-based CDT at time T;
5. **let** S_T be the set of units where the ICI-based CDT detects a change at time T;
6. **if** (S_T is not empty) {
7. *Each unit in* S_T: run the refinement procedure, sent $T_{ref,i}$ to the aggregation center.
8. *Aggregation center*: compute T_{ref} out of $T_{ref,i}$, $1 \le i \le N$, send T_{ref} to each unit.
9. *Each unit*: send features in $[T_{ref}, T]$ to the aggregation center;
10. *Aggregation center*: runs the hypothesis test;
11. **if** (second-level test detects a change){
12. Change is validated.
13. *Each unit in* S_T the ICI-based CDT is re-trained on the new process status}
14. **else**{
15. Change hypothesis is discarded (false positive);
16. *Each unit in* S_T: reconfigure the ICI-based CDT to improve its performance }}}

unit independently assesses whether the incoming data have been generated by the process in its initial (stationary) state or not by relying on the ICI-based CDT. When no units detect a change, the process is considered stationary, and each unit waits for the next measurements.

Denote as S_T the set of units detecting a variation at time T (line 5): each unit in S_T runs independently the refinement procedure described in [12] to provide an estimate $T_{ref,j}$, $1 \le j \le |S_T|$ of the change time-instant T^* (line 7). These estimates are sent to the aggregation center and there processed (e.g., by choosing the earliest one, or by computing their mean) to determine a common unique estimate T_{ref} of T^* (line 8).

The estimated change time T_{ref} is then propagated through the network and made available to units. In turn, units send the locally stored extracted features in the $[T_{ref}, T]$ temporal interval (line 9) to the data-aggregation center (an internal shift buffer is made available to locally store incoming observation).

At the aggregation center the hypothesis test aims at reducing false positives by performing a multivariate analysis to assess if there is a statistical evidence that features computed in O_{i,T_0} differs from features in $[T_{ref}, T]$, after the suspected change (line 10). If the aggregation center validates the detection (line 12), each node is retrained to monitor further variations w.r.t. the new process state on the forthcoming observations (line 13). In particular, each node restarts the ICI-based CDT using the features computed in $[T_{ref}, T]$. On the contrary, when the aggregation center does not validate the detection (line 15), each units in S_T is informed to have provided a false positive, and it is retrained using the features estimated from the initial training set O_{i,T_0}, to keep on monitoring changes w.r.t. the initial status (line 16). We provide now further details and comments related to the proposed algorithm.

3.1 ICI-based change-detection test at the unit level

Since in most of practical applications the distribution of the process before and after the change remains unknown, nonparameteric sequential CDTs have to be enforced at the unit level. Among available solutions present in the literature, e.g., see [2][3], the ICI-based change-detection test [10] has been selected for its low computational complexity and strong theory. Moreover, the test, differently from other nonparameteric sequential CDT mechanisms, is endowed with a change-detection refinement procedure able to provide an accurate estimate T_{ref} of the time instant T^*; such an information is then used in Algorithm 1 (line 7).

Basically, the ICI-based CDT monitors the data generating process by relying on i.i.d. and Gaussian distributed features (in the stationary case); the Gaussian hypothesis is satisfied thanks to ad-hoc transformations. The features we use here are derived from the sample mean and variance (computed on disjoint subsequences of observations) and represent the data transmitted to the aggregation center to validate the unit-level detections (line 9).

3.2 The Change-Detection Test at the Data-Aggregation Center

The test running at the aggregation center level aims at assessing the detections raised by the sensing-units by exploiting group information. In particular, the T_{ref} obtained by processing the local estimates $T_{ref,i}$ allows us for selecting the observations in $[T_{ref},T]$ stored in each sensing unit, that are representative of the process after the change. A multivariate hypothesis test based on the Hotelling T-square statistic [11], which is the classical technique to inference a change in the mean of a Gaussian multivariate random variable, is executed to assess if there is a statistical evidence (according to a defined significance level α) that the mean features value in O_{T_0} equals that in $[T_{ref},T]$. The analysis performed at the aggregation center focuses on the feature detecting the change at the sensing units: let F^0_i and F^1_i be the sequences of feature values at the i^{th} sensing unit in O_{i,T_0} and $[T_{ref},T]$, respectively; let n_0 and n_1 be their lengths. The inference is performed on the mean vectors $\bar{F}^0$and $\bar{F}^1$, defined such that their i^{th} components are $(\bar{F}^0)_i = \sum_{j=1}^{n_0} F_i^0(j)/n_0$ and $(\bar{F}^1)_i = \sum_{j=1}^{n_1} F_i^1(j)/n_1$,with i=1,..,N. The covariance matrix Σ is computed by pooling the covariances estimated from the features F^0_i and from F^1_i. The Hotelling T-square statistic is

$$S = (\bar{F}^0 - \bar{F}^1)' \left(\left(\frac{1}{n_0} + \frac{1}{n_1} \right) \Sigma \right)^{-1} (\bar{F}^0 - \bar{F}^1),$$

and is distributed as

$$\left(\frac{n_0+n_1-2}{n_0+n_1-N-1} \right) \mathcal{F}(N, n_0 + n_1 - N - 1),$$

where $\mathcal{F}$ denotes the F distribution. This allows us for verifying if the null hypothesis "the difference between $\bar{F}^0 - \bar{F}^1$ equals 0" needs to be rejected with confidence α.

The Hotelling T-square statistics require $n_0>N$ and $n_1>N$. For this reason, the length of the training sequence and the minimum amount of samples to be considered between T_{ref} and T should be suitably adapted to the number of sensing units in the network. Large networks could be eventually partitioned, only for change-detection purposes, into smaller clusters of units.

Table 1. Simulations results.

			Proposed CDS			**Traditional *k/N* CDS**		
			***α*=0.05**	***α*=0.01**	***α*=0.005**	***k*=1**	***k*=5**	***k*=10**
D1	**Abrupt** δ =.1σ	**FP(%)**	16.0	4.6	4.0	69.3	0	0
		FN(%)	0	0.6	2.6	0	0	0
		MD	4456.6	5289.5	5350.2	4618.9	8523.4	14508.2
	Abrupt δ=.5σ	**FP(%)**	16.0	4.6	4.0	69.3	0	0
		FN(%)	0	0	0	0	0	0
		MD	808.8	793.8	794.2	852.6	1478.1	2179.0
	Abrupt δ =2σ	**FP(%)**	16.0	4.6	4.0	69.3	0	0
		FN(%)	0	0	0	0	0	0
		MD	449.1	449.1	449.2	442.6	468.9	545.4
	Drift δ= .1σ	**FP(%)**	16.0	4.6	4.0	69.3	0	0
		FN(%)	5.3	15.3	23.3	0	0	4.6
		MD	34884.8	39642.7	41709.4	21767.1	37743.6	51967.9
	Drift δ = .5σ	**FP(%)**	16.0	4.6	4.0	69.3	0	0
		FN(%)	0	0	0	0	0	0
		MD	11633.7	12215.4	12861.3	10034.3	14937.8	19263.2
	Drift δ = 2σ	**FP(%)**	16.0	4.6	4.0	69.3	0	0
		FN(%)	0	0	0	0	0	0
		MD	5019.6	4997.4	5181.3	4624.2	6972.7	8703.8
D2	**Abrupt**	**FP(%)**	11.3	2.0	0.6	65.3	0	0
		FN(%)	0	0	0	0	0	0
		MD	461.5	463.8	463.8	457.8	555.9	2643.1

4 Experiments

To validate the effectiveness of the suggested distributed change-detection test we considered both synthetic and real applications applied to a network of N=10 sensorial units. Performances of the proposed approach are compared with those of traditional methods (a global detection is raised when at least k sensing units out of N detect a variation [5]-[7]). At the unit level the detection test is the ICI-based CDT shown to be particularly effective in the literature [10]. Three indexes have been considered to assess the performances of the tests:

- False positive index (FP): it counts the number of times a test detects a change when there it is not.
- False negative index (FN): it counts the times a test does not detect a change when there it is.
- Mean Delay (MD): it represents the time delay in detecting a change (expressed in terms of the number of observations).

Application D1 – The i^{th} sensing unit receives 90000 observations extracted from a Gaussian distribution $N(\mu = 1, \sigma^2 = 1)$ then, to reproduce a scenario where each unit has different gain and offset values, these data are scaled by σ_i>0 (the gain), and a constant term μ_i>0 (the offset) is added. Thus, the data generating process at each unit can be described as $N(\mu_i + \sigma_i, \sigma_i^2)$. We considered two kinds of perturbations having magnitude $\delta_i \in \{0.1\sigma_i, 0.5\sigma_i, 2\sigma_i\}$ affecting the mean value at sample T^*=30000: an abrupt change, where the mean suddenly increases of δ_i, and a drift, where the mean increases linearly starting at T^* and reaches $\mu_i + \delta_i + \sigma_i$ at T=90000.

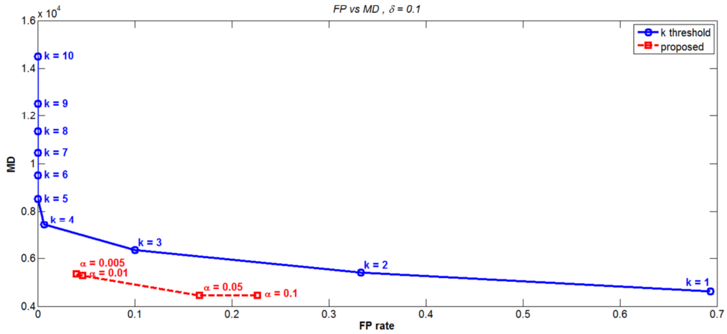

Fig. 1. Mean Delay (MD) w.r.t. false positives for the proposed CDT (with α ranging from 0.005 to 0.1) and the traditional k out of N detection test with $\delta = 0.1\sigma$.

Application D2 – Each sensing unit is a photodiode excited by X-rays. Sequences contain 60000 samples and show a perturbation affecting the mean at sample 30000 in the $\delta \in \{0.1\sigma, 2\sigma\}$ range (being σ estimated from samples [0,30000]). As shown in [8], these data are far from being Gaussian distributed.

The ICI-based CDTs in execution on the units have been trained with the first 400 samples for each dataset. According to [12], we experimentally fixed Γ=2.5 and Γ_{ref}=3 (the tuning parameters for the ICI-based CDT and for the refinement procedure, respectively). We considered α=0.005, α=0.01 and α=0.05 for the second-level hypothesis test (1-α is the test confidence).

Table I shows the comparison between the proposed and the traditional approach with $k \in \{1, N/2, N\}$; performances are averaged over 150 runs. As far as application D1-abrupt is concerned, the proposed CDT provides prompter detections w.r.t. the traditional approach, yet guaranteeing lower false positives. As expected, as confidence 1-α decreases, the CDT increases both false positives and detection delays. The same behavior can be observed when parameter k in the traditional approach. Obviously, the detection delays of both approaches decrease as the intensity of the perturbation δ increases, since the change is more easily detectable. These results are particularly interesting since the proposed approach allows to detect a change in the data generating process even when the variation is detected by just a sensing unit (and then confirmed at the aggregation center), while in the traditional approach the change must be detected by at least k units (which could be critical when the change in the process is perceived only by a subset of the N units).

Experiments on application D1-drift show that both false negatives and the mean delay are higher than those in the abrupt change case since the change is smooth and more difficult to detect. In particular, with $\delta = 0.1\sigma$, the suggested approach provides lower performance than the traditional approach since, as shown in [8], the aggregation center might discard the detections provided by the sensing units due to the lack of statistical evidence in rejecting the stationary hypothesis (this might be caused both by the small magnitude of the perturbation and by an inaccurate estimate of T_{ref}). To reduce both false positives and detection delays, higher values of α should be considered (e.g. $\alpha \geq 0.1$). On the contrary, the proposed approach overcomes the performance of the traditional methods for larger values of δ. Experimental results of application D2 show that the proposed CDT well behaves even with real non-Gaussian data and that results are in line with those of application D1.

A more detailed comparison between the proposed and the traditional approaches has been performed to evaluate performances when k ranges from 1 to N. Results are given in Figure 1 for application D1-abrupt with $\delta = 0.1\sigma$. Performance improvements are appreciable, obtained at the expenses of negligible increases of false negatives as presented in Table 1: in particular, given a tolerated percentage of false positives, the proposed approach guarantees a lower detection delay once compared to the traditional one. Similarly, at equal values of detection delays, the proposed approach provides lower false positive rates. For example, considering acceptable a FP rate of 5%, the suggested test yields MDs about 25% lower than the traditional approach.

5 Conclusions

This work presents a distributed nonparametric CDT designed to work in networked embedded systems and sensor/actuator networks. The novelty of the proposed approach resides in the distributed two-levels CDT where a change is first detected (ICI-based method) at the unit level and then assessed at the cluster level by exploiting cluster information (Hotelling test). This allows the test for reducing false positives, a common problem which arises in sequential CDTs. Experiments applied to synthetic and real applications show the effectiveness of the proposed approach.

References

1. Chiang, L.H., Russell, E.L., Braatz, R.D.: Fault Detection and Diagnosis in Industrial Systems. Springer, London (2001)
2. Lai, T.L.: Sequential analysis: some classical problems and new challenges. Statistica Sinica. 11(2), 303 – 350 (2001)
3. Basseville, M., Nikiforov, I. V.: Detection of Abrupt Changes: Theory and Application. Prentice-Hall, N.J. USA (1993)
4. Mei, Y.: Information bounds and quickest change detection in decentralized decision systems. Information Theory, IEEE Trans. on, 51(7), 2669 – 2681 (2005)
5. Tartakovsky, A., Veeravalli, V.: Quickest change detection in distributed sensor systems. In: 6th Int. Conf. on Information Fusion, 756 – 763 (2003)
6. Tartakovsky, A., Veeravalli, V.: Asymptotically optimal quickest change detection in distributed sensor systems. Sequential Analysis, 27(4), 441 – 475 (2008)
7. Yang, D., Qi, H.: An Effective Decentralized Nonparametric Quickest Detection Approach. In 20th Int. Conf. on Pattern Recognition, pp. 2278 – 2281 (2010)
8. Alippi, C., Boracchi, G., Roveri, M.: A hierarchical nonparametric sequential change-detection test. In: IEEE 2011 International Joint Conference on Neural Networks (2011)
9. Ljung L.: System identification: Theory for the user. Prentice Hall, N.J. USA (1999)
10. Alippi, C., Boracchi, G., Roveri, M.: Change Detection Tests Using the ICI rule. In: IEEE 2010 International Joint Conference on Neural Networks, pp. 1 – 7 (2010)
11. Johnson, R. A., Wichern, D.W.: Applied multivariate statistical analysis, Prentice Hall, N.J. USA (1998)
12. Alippi, C., Boracchi, G., Roveri, M.: Adaptive Classifiers with ICI-based Adaptive Knowledge Base Management. In: Proc 20th international conference on Artificial neural networks: Part II. LNCS, vol. 6353, pp. 458 – 467. Springer-Verlag (2010)