Méthode à base de patterns pour la détection d'anomalies

Inès Ben Kraiem, Faiza Ghozzi, André Péninou, Olivier Teste

Méthode basée sur les patterns pour la détection simultanée d'anomalies multiples dans les réseaux de capteurs

Ines Ben Kraiem [1], Faiza Ghozzi [2], Andre Peninou [1], Olivier Teste [1]

1. *Université de Toulouse, UT2J, IRIT, Toulouse, France*
   {ines.ben-kraiem,andre.peninou,olivier.teste}@irit.fr
2. *Université de Sfax, ISIMS, MIRACL, Sfax, Tunisia*
   faiza.ghozzi@isims.usf.tn

RÉSUMÉ. *La détection d'anomalies dans les applications réelles de distribution de fluide est une tâche difficile, en particulier lorsque l'on cherche à détecter simultanément différents types d'anomalies. La résolution de ce problème est importante dans plusieurs domaines par exemple, dans les applications de gestion et de supervision de bâtiments. Dans cet article, nous présentons l'algorithme CoRP "Composition of Remarkable Points", une approche configurable basée sur la modélisation de patterns de détection simultanée d'anomalies multiples. CoRP applique un ensemble de patterns, défini par l'utilisateur, afin d'annoter (labels) les points remarquables dans une série temporelle uni-variée, puis détecte les anomalies par composition de labels. En comparant avec des algorithmes de la littérature, notre approche se montre plus robuste et plus précise pour détecter tous les types d'anomalies observées dans des déploiements réels. Nos expérimentations reposent sur des données du monde réel et des données de benchmark issues de la littérature.*

ABSTRACT. *The detection of anomalies in real fluid distribution applications is a difficult task, especially, when we seek to simultaneously detect different types of anomalies and possible sensor failures. Resolving this problem is increasingly important in many areas, for example, in building management and supervision applications. In this paper we introduce CoRP "Composition of Remarkable Points" a configurable approach based on pattern modelling, for the simultaneous detection of multiple anomalies. CoRP evaluates a set of patterns that are defined by users, in order to tag the remarkable points (labels) in a univariate time series and then detects among them the anomalies by composition of labels. By comparing with literature algorithms, our approach appears more robust and accurate to detect all types of anomalies observed in real deployments. Our experiments are based on real-world data and benchmark data from the literature.*

MOTS-CLÉS : *Réseaux de capteurs, Détection d'anomalies, Méthode basée sur les patterns.*

KEYWORDS: *Sensor networks, Anomaly detection, Pattern-based method.*

1. Introduction

Les réseaux de capteurs jouent un rôle important dans la supervision et l'exploration des réseaux de distribution de fluides (e.g., énergie, eau, chauffage) à l'échelle d'un campus et plus largement d'une ville ou d'une région. L'exploitation de ces réseaux repose sur des données relevées par des capteurs. Ces données comportent des anomalies qui nuisent à la supervision (e.g., fausses alarmes, arrêts). Par exemple, la figure 1 illustre un changement brusque (représenté par une croix) dans les mesures de capteurs, engendrant un changement de niveau permanent suite à un problème de matériel (e.g., capteurs endommagés, changement de capteur). Les triangles illustrent plusieurs pics représentant des défauts de lecture liés à un événement imprévu (e.g., panne, rupture). Enfin, le rectangle représente un décalage constant dans les mesures (dû à un problème de communication). Dans un tel cas de figure, les mesures du capteur peuvent différer de leurs valeurs réelles ou attendues et se transformer ainsi en anomalies ce qui rend la tâche d'exploitation plus difficile et complexe. C'est dans cette optique, que la détection d'anomalies apparaît comme étant le moyen pour identifier les évènements anormaux et détecter des comportements qui ne sont pas conformes au comportement attendu (Chandola *et al*., 2009). Au-delà de la supervision des réseaux de capteurs, il existe un large éventail d'applications pour lesquelles il est primordial de détecter les anomalies pour faciliter l'analyse des données notamment la détection d'intrusions, la détection de dommages industriels, la détection d'anomalies dans l'imagerie médicale, la détection d'anomalies dans les données textuelles, la surveillance, la détection de fraude dans les transactions financières, etc (Hodge, Austin, 2004). Plusieurs techniques ont été proposées dans la littérature et classées selon les domaines d'applications ou les types d'anomalies à détecter (Chandola *et al*., 2009). Néanmoins, ces techniques ne permettent pas toujours de détecter tous les types d'anomalies, obligeant les applications à utiliser plusieurs méthodes pour détecter de multiples anomalies de nature diverse.

Cet article se place dans le cadre d'applications réelles ayant des anomalies spécifiques au métier (gestion des fluides sur le campus de Rangueil-Toulouse). La problématique est de trouver une méthode permettant de détecter de multiples anomalies de différents types observées lors de déploiements réels tout en maximisant le nombre d'anomalies détectées et en minimisant les erreurs. Nous traitons dans ce contexte des séries temporelles uni-variées. La difficulté de disposer d'une technique robuste pour détecter l'ensemble des anomalies nous amène à définir une nouvelle méthode configurable nommée **CoRP** "Composition of Remarkable Points". Cette méthode permet, premièrement, de détecter des points qui paraissent remarquables dans les séries temporelles en évaluant des patterns et, deuxièmement, de créer des compositions de points remarquables utilisées pour identifier de multiples anomalies.

La suite de cet article est structurée comme suit. Dans la section 2, nous étudions quelques techniques et algorithmes de la littérature traitant la détection d'anomalies dans les séries temporelles. Dans la section 3, nous décrivons et détaillons notre approche. La section 4 présente les expérimentations effectuées, d'une part, sur notre

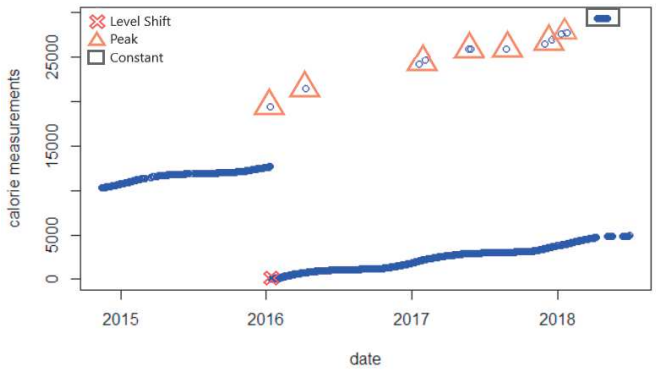

Figure 1. Exemple d'anomalies dans les mesures de capteurs.

étude de cas réel et, d'autre part, sur des données de référence. Enfin, nous concluons avec les perspectives et les recherches ultérieures possibles pour notre travail.

2. État de l'art sur la détection d'anomalies

La plupart des recherches existantes en détection d'anomalies portent sur divers domaines d'applications (Chandola *et al*., 2009) (Hodge, Austin, 2004) (Sreevidya *et al*., 2014). Ces articles identifient de nombreuses techniques de détection d'anomalies en fonction du domaine d'application. Les principales approches sont basées sur le regroupement, la classification, les statistiques, les voisins les plus proches, la régression, la décomposition spectrale et la théorie de l'information.

Sharma *et al*. (2010) ont exploré des techniques pour détecter des anomalies courtes, de bruits et constantes. Pour cela, quatre classes de méthodes de détection d'anomalies sont proposées : méthodes basées sur des règles, méthodes d'estimation des moindres carrés, méthodes par apprentissage (HMM) et méthodes basées sur l'analyse des séries temporelles (ARIMA). Bien que chacune de ces méthodes détecte les types d'anomalies spécifiques, elles génèrent toujours des erreurs, en particulier dans un contexte d'anomalies multiples. Pour cette raison, des méthodes hybrides ont été proposées, Hybrid (U) et Hybrid (I), afin de réduire le nombre de faux négatifs et de faux positifs en combinant les résultats de chaque méthode (Sharma *et al*., 2010).

Yao *et al*. (2010) proposent un algorithme appelé analyse de séquence segmentée (SSA) qui consiste à comparer les mesures à évaluer avec une série temporelle de référence. Cette méthode ne permet pas de détecter avec précision toutes les anomalies. Pour cette raison, les auteurs ont proposé une approche hybride qui est une combinaison de SSA et de la méthode basée sur les règles. Ainsi, leur approche consiste à appliquer la méthode basée sur les règles dans une première phase afin de détecter les anomalies courtes, puis l'algorithme SSA pour détecter les anomalies restantes.

Il existe d'autres méthodes de détection d'anomalies sur des données unies-variées telles que le test ESD généralisé (Extreme Studentized Deviate) (Rosner, 1983) et Change Point (Basseville *et al*., 1993) (Aminikhanghahi, Cook, 2017). L'algorithme ESD utilise des fonctions statistiques telles que la moyenne et la déviation standard pour la détection d'anomalies. ESD nécessite de spécifier une limite supérieure pour le nombre probable d'anomalies existantes ; ceci n'est pas possible pour toutes les applications. La méthode de changement de point (Change Point) détecte les changements de distribution (e.g., moyenne, variance, covariance) dans les mesures du capteur (Basseville *et al*., 1993). Cette méthode détecte chaque changement sous forme d'anomalies.

D'autres méthodes sont basées sur la technique du voisin le plus proche pour la détection d'anomalies et peuvent être regroupées en deux catégories (Chandola *et al*., 2009) : (1) les techniques qui utilisent la distance d'une instance de données à son kème voisin le plus proche comme score d'anomalie (Upadhyaya, Singh, 2012).
(2) les techniques qui calculent la densité relative de chaque instance de données pour calculer son score d'anomalies, par exemple, l'algorithme LOF (Local Outlier Factor) (Breunig *et al*., 2000).

Bien que chacune de ces méthodes ait été conçues pour la détection d'anomalies, nous pensons qu'elles ne satisfont pas toutes les propriétés souhaitables, y compris la détection de multiples types d'anomalies observées de manière simultanée dans les déploiements réels avec un taux d'erreurs faible. De plus, plusieurs méthodes parmi les méthodes mentionnées nécessitent un prétraitement ou un post-traitement à travers des méthodes hybrides pour améliorer leurs résultats. Pour évaluer les performances de ces méthodes, nous avons sélectionné des algorithmes appartenant à différentes techniques et proches de la détection des types d'anomalies recherchées. Nous effectuons une comparaison entre ces méthodes appliquées sur notre étude de cas dans la section 4.

Exploration des méthodes de détection existantes. Dans notre étude, nous avons exploré cinq méthodes appartenant à quatre techniques différentes pour détecter les types d'anomalies observées dans notre application.

– Méthode basée sur les règles : nous avons utilisé deux règles pour détecter les anomalies courtes (changement anormal) et les anomalies constantes (pas de variation) (Sharma *et al*., 2010). *La règle d'anomalie courte* traite la série temporelle en comparant à chaque fois deux observations successives : on détecte une anomalie si la différence entre ces observations est supérieure à un seuil donné. Pour déterminer automatiquement le seuil de détection, nous avons utilisé l'approche basée sur l'histogramme (Ramanathan *et al*., 2006). *La règle d'anomalie constante* calcule l'écart-type pour un ensemble d'observations successives. Si cette valeur est égale à zéro l'ensemble est déclaré comme une anomalie.

– Méthode basée sur la densité : cette approche consiste à comparer la densité autour d'un point par rapport à la densité de ses voisins locaux. Breunig *et al*. (2000) ont proposé l'algorithme LOF. Dans cette méthode, les scores des anomalies sont mesurés en utilisant un facteur de valeur aberrante local, qui est le rapport entre la

densité locale autour de ce point et la densité locale autour de ses plus proches voisins. Le point dont la valeur LOF est élevée est déclaré une anomalie.

– Méthode basée sur les statistiques : premièrement, nous avons utilisé la méthode ESD pour la détection automatique des anomalies locales et globales. Deuxièmement, nous avons utilisé la méthode Change Point pour détecter le changement de niveau.

– Méthode basée sur l'analyse des séries temporelles : le principe de cette approche est d'utiliser les corrélations temporelles pour modéliser et prédire les valeurs de la série temporelle. Nous avons utilisé le modèle ARIMA (AutoRegressive Intergrated Moving Average) pour la création du modèle de prédiction selon l'approche décrite par Chen et Liu (1993). Une mesure de capteur est comparée à sa valeur prédite pour déterminer si elle est une anomalie.

Il existe des implémentations open source pour des algorithmes tels que LOF, ARIMA, S-H-ESD et Change Point, ((Hochenbaum *et al.*, 2017) (Cleveland *et al.*, 1990) (Rosner, 1983) (Aminikhanghahi, Cook, 2017)) que nous avons utilisés pour les expérimentations. En revanche, nous avons mis en œuvre d'autres approches (règle courte et règle constante) en fonction des sources disponibles.

Le tableau 1 représente la synthèse des méthodes que nous avons explorées pour détecter chaque type d'anomalies détectées lors de déploiements réels et présentées dans la figure 1.

Tableau 1. Les méthodes de détection d'anomalies étudiées.

Type d'anomalies	Méthodes de détection
Changement anormal	Règle courte, $ARIMA$, LOF, $S-H-ESD$
Valeurs constantes	Règle constante
Changement de niveau	$ARIMA$, Change Point

3. Méthodologie CoRP pour la détection simultanée d'anomalies multiples

En situation d'exploitation réelle, la supervision des réseaux de capteurs se fait par les experts (ingénieurs d'exploitation, techniciens de maintenance, etc) en observant les courbes afin de détecter les points qui semblent remarquables et qui montrent des comportements inhabituels. Typiquement, ce sont les mesures des capteurs de notre étude de cas illustrées figure 1. Ces points remarquables sont les variations inhabituelles entre les points successifs d'une série temporelle et qui sont les marqueurs (ou indices) de possibles anomalies. Afin de mettre en œuvre cette démarche de détection de façon automatique, nous avons créé notre approche configurable, appelée CoRP, de détection d'anomalies. Elle est basée sur des patterns pour la détection des points remarquables et sur des compositions de ces points afin d'identifier les anomalies.

3.1. Notations utilisées

Définition 1 *Une série temporelle* est composée d'observations ou de points successifs collectés séquentiellement dans le temps à intervalle régulier. Ces points repré-

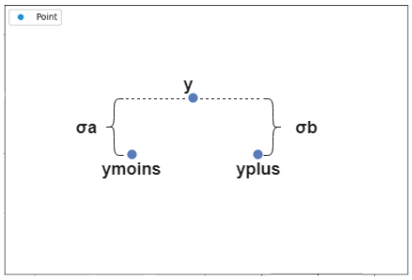

FIGURE 2. *Labellisation d'un point remarquable "y" par un pattern σ_a et σ_b.*

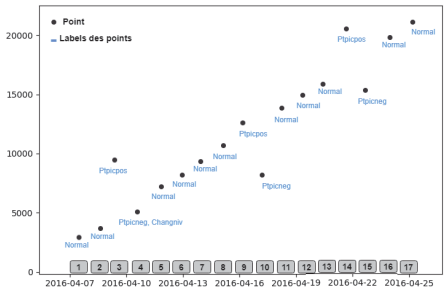

FIGURE 3. *Labellisation d'une série de points remarquables (algorithme 1).*

sentent les mesures associées à un horodatage indiquant l'heure de sa collecte. Soit $Y_i = \{y_1, y_2, y_3....\}$ une série temporelle représentant la séquence des points ou mesures de capteurs collectées pour chaque point i $\in \mathbb{N}$.

Définition 2 *Un point (ou mesure)* est composé d'une valeur et d'un horodatage. Dans cet article, on note un point $y_i = (t_i, v_i)$ tel que t_i est l'horodatage de y_i (noté $t(y_j)$) et v_i est la valeur de y_i (noté $v(y_j)$).

3.2. Description de CoRP

En s'appuyant sur l'expérience des experts (détection des points remarquables puis identification des anomalies), l'algorithme CoRP est construit en deux phases. La première consiste à détecter et annoter les points considérés comme remarquables dans la série temporelle. La deuxième phase consiste à créer une composition de points remarquables permettant d'identifier les anomalies.

3.2.1. Détection des points remarquables

La détection des points remarquables est réalisée à partir des patterns de détection.

Définition 3 *Un pattern p* est défini par un triplet $p = (l, \sigma_a, \sigma_b)$ où l est un label qui étiquettera un point détecté comme remarquable par le pattern, σ_a et σ_b sont deux seuils utilisés pour décider si un point donné est remarquable. Un pattern est appliqué sur trois points consécutifs y_{j-1}, y_j, y_{j+1} d'une série temporelle Y, points qui sont notés y_{moins}, y, y_{plus}. σ_a correspond à l'écart entre $v(y_{moins})$ et $v(y)$ tandis que σ_b correspond à l'écart entre $v(y)$ et $v(y_{plus})$ comme illustré figure 2. Lorsqu'un pattern est vérifié sur y_{moins}, y, y_{plus}, le label du pattern est utilisé pour étiqueter le point y.

Définition 4 *Une série temporelle labellisée* est une série temporelle de points sur lesquels sont ajoutés les labels détectés par les patterns.

Définition 5 *Un point remarquable* y_i d'une série temporelle labellisée est défini par un triplet (t_i,v_i, L_i) où t_i est son horodatage, v_i est sa valeur et $L_i = \{l_1, l_2, ...\}$ est une liste de labels caractérisant le point.

Les patterns sont donc utilisés pour détecter les points remarquables et leurs ajouter le label correspondant. Ainsi, la liste des labels d'un point remarquable est constituée des labels des différents patterns vérifiés sur ce point. La figure 3 illustre un extrait d'une série temporelle labellisée des données d'index qui ont tendance à croître. Ainsi, la figure 3 comporte 4 exemples de labels (Normal, Ptpicpos, Ptpicneg, Changniv). Notons le point 4 qui comportent 2 labels.

Exemple. Des exemples de patterns utilisés afin de labelliser la courbe de la figure 3 sont : (i) un "Point Pic Positif" remarquable (Ptpicpos, 100, 100) où Ptpicpos représente le label descriptif du pattern, σ_a=100 et σ_b=100 ; (ii) un "Point Pic Négatif" remarquable (Ptpicneg, -100, -100) ; et (iii) un "Changement de Niveau" remarquable (Changniv, -1000, -100).

L'algorithme 1, appelé EvaluatePattern, permet d'évaluer un pattern à l'aide de règles. Cette fonction prend en entrée trois points successifs notés y_{moins}, y et y_{plus} et le pattern à évaluer, et renvoie le résultat de l'évaluation. Différentes règles de vérifications sont appliquées en fonction des signes de σ_a et σ_b. Les règles pour comparer y_{moins} et y en fonction de σ_a sont les suivantes : (i) Si $\sigma_a > 0$, la règle à vérifier est $v(y) >= v(y_{moins}) + \sigma_a$; (ii) Si $\sigma_a < 0$, la règle à vérifier est $v(y) <= v(y_{moins}) + \sigma_a$; et (iii) Si $\sigma_a = 0$, la règle à vérifier est $v(y) = v(y_{moins})$. Les règles pour comparer y et y_{plus} en fonction de σ_b sont similaires (voir l'Algorithme1).

Algorithm 1 Evaluation d'un pattern

function BOOLEAN EVALUATEPATTERN($y_{moins}, y, y_{plus}, p$)
Input $y_{moins}, y, y_{plus}, p = (l_p, \sigma_a, \sigma_b)$
Output Boolean
  if $p.\sigma_a > 0$ **then** leftRemarkable<-($v(y) \geq v(y_{moins}) + p.\sigma_a$? true : false)
  else if $p.\sigma_a < 0$ **then** leftRemarkable<-($v(y) \leq v(y_{moins}) + p.\sigma_a$? true : false)
  else if $p.\sigma_a = 0$ **then** leftRemarkable<-($v(y) = v(y_{moins})$? true : false)
  end if
  if $p.\sigma_b > 0$ **then** rightRemarkable<-($v(y) \geq v(y_{plus}) + p.\sigma_b$? true : false)
  else if $p.\sigma_b < 0$ **then** rightRemarkable<-($v(y) \leq v(y_{plus}) + p.\sigma_b$? true : false)
  else if $p.\sigma_b = 0$ **then** rightRemarkable<-($v(y) = v(y_{plus})$? true : false)
  end if
return (leftRemarkable and rightRemarkable)
end function

L'algorithme 2 fait appel à la fonction EvaluatePattern pour traiter une série temporelle. Il prend en entrée la série temporelle initiale et la liste des patterns et renvoie une nouvelle série temporelle labellisée. Le traitement consiste à parcourir la série temporelle et la liste des patterns. Pour chaque point et pour chaque pattern p, la fonction EvaluatePattern est appelée afin d'ajouter (ou pas) le label de p au point évalué.

Algorithm 2 Detection de points remarquables

Input $Y = \{y_1, y_2, y_3....\}$, $P = \{p_1, p_2, p_3....\}$
Output Y_L série temporelle labellisée

for i in range(2..|Y|-1) **do**
  for k in range(1..|P|) **do**
    if EvaluatePattern(y_{i-1},y_i,y_{i+1},p_k) **then**
      L_i <- $L_i + p_k.l$
    end if
  end for
end for
  return Y_L

3.2.2. Composition de patterns

Le résultat de l'algorithme 2 est une série temporelle labellisée à l'aide de patterns. Comme montré dans la figure 4, un point peut être labellisé par un ou plusieurs patterns. A partir d'un sous-ensemble de points d'une série temporelle labellisée, on construit par concaténation des labels L_i de ces points remarquables, une chaîne de labels. Cette chaîne de labels est utilisée pour détecter une anomalie. Une anomalie est reconnue par une composition de labels et par la vérification de conditions sur les valeurs des points.

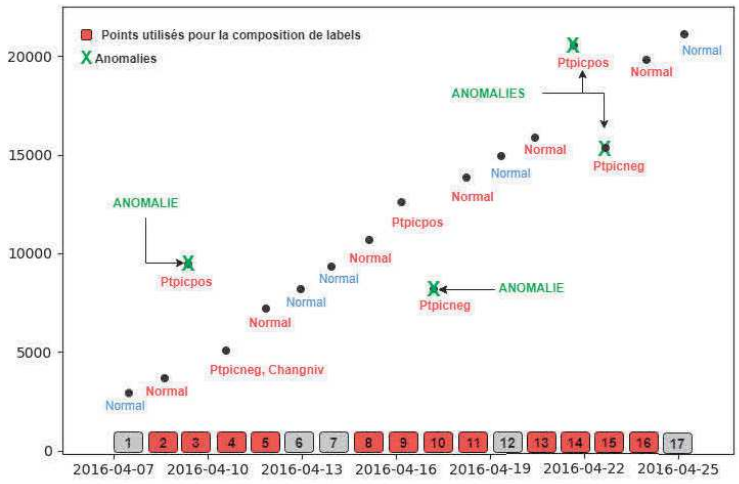

Figure 4. Résultat de la phase 2 de l'algorithme CoRP.

Définition 6 *Une anomalie* est un ou plusieurs points remarquables appartenant à un sous-ensemble de points pour lequel sont vérifiées, d'une part, une composition des labels de ces points et, d'autre part, une condition exprimée sur les valeurs de ces points. L'anomalie est enfin identifiée sur un ou plusieurs points de cette composition.

Pour définir une composition de labels, nous proposons une grammaire, illustrée dans la figure 5, qui définit les éléments d'une composition de labels. La grammaire permet de définir les labels possibles (un ou plusieurs) sur des points successifs permettant de reconnaître une composition de labels. La grammaire part des labels posés sur les points (<label>). Les labels peuvent être combinés sur un seul point avec des expressions logiques AND, OR et NOT (<label-comp> and <point-label>). Par exemple "l1 AND NOT l2 AND l3" désigne un point labellisé avec l1, labellisé avec l3 et non labellisé l2. Chaque composition de labels sur un point unique peut être répétée sur des points successifs par des quantificateurs : ?, + et * (<label-enum>). Par exemple "(l1) +" signifie que la composition doit comporter un ou plusieurs points successifs labellisés l1. La composition de labels finale est créée à travers une succession de labels séparés par "." (< composition >). Par exemple, "l1. (l2) *. l1 OR l3" signifie un point labellisé par l1 suivi de zéro ou plus de points labellisés par l2 suivi d'un point labellisés par l1 ou par l3.

```
<composition> ::= <label-enum> ("." <label-enum>) *

<label-enum> ::= <label-comp>
           |   "(" <label-comp> ")" "?"
           |   "(" <label-comp> ")" "*"
           |   "(" <label-comp> ")" "+"

<label-comp> ::= <point-label> ("OR" <point-label>) *
           |     <point-label> ("AND" <point-label>) *
           |     <point-label>

<point-label> ::= <label>
           |      "NOT" <label>

<label> ::= list of labels defined by patterns
```

Figure 5. Grammaire pour la définition d'une composition de labels.

Définition 7 *Une composition de labels* permettant de reconnaître une anomalie est composée de trois parties :

– *composition* : la composition des labels des points remarquables qui est une séquence de points comportant des labels définies selon la grammaire présentée figure 5 ;

– *condition* : une condition entre les valeurs des points reconnus (ceux correspondants à la séquence des labels). Une même composition de labels peut correspondre à différentes anomalies. Cette condition est créée à l'aide des opérateurs (<, <=, etc) permettant de comparer des valeurs et des opérateurs logiques (AND/OR/NOT) permettant de combiner des comparaisons. Afin d'éviter l'utilisation de la notation $v(y_i)$, nous notons par v_i la valeur du ième point reconnu par la composition, v_1 le premier et v_n le dernier ; notons que le nombre de points impliqués dans la composition peut être variable compte tenu des quantificateurs utilisables dans la composition ;

– *conclusion* : l'anomalie identifiée pour laquelle sont précisés son type (nom de l'anomalie) et la liste des valeurs (points) où se situe l'anomalie détectée.

A titre d'exemple, nous donnons les compositions de labels pour identifier trois types d'anomalies : (i) anomalie de valeurs constantes, (ii) anomalie de valeurs en pic négatif, et (iii) anomalie de valeurs en pic positif ; cette dernière est possiblement reconnue à partir de 2 compositions de labels.

Label-composition 1
composition : Normal . Ptpicpos . Ptpicneg . Normal
condition : $v_2 > v_4$ and $v_3 > v_1$
conclusion : positive peak -> v_2
Label-composition 2
composition : Normal . Ptpicpos . Ptpicneg . Normal
condition : $v_2 < v_4$ and $v_3 < v_1$
conclusion : negative peak -> v_3
Label-composition 3
composition : Begincstpos . Cst* . Endcstneg
condition : $v_1 == v_2$ and $v_{n-1} == v_n$
conclusion : constant -> all
Label-composition 4
composition : Normal . Ptpicpos . Ptpicneg AND Changnivneg . Normal
condition : $v_2 > v_4$ and $v_3 > v_1$
conclusion : positive peak -> v_2

Exemple. Considérons les sous-ensembles de points présentés sur la figure 4 en rouge. Les points d'indices 2 à 5 donnent la série de labels suivante : (Normal . Ptpicpos . Ptpicneg and Changniv . Normal) détectée par la composition de labels 4 de l'exemple. Cette composition permet donc de détecter l'anomalie de pic positif en 3. Les points d'indices 8 à 11, déclenchent les compositions de labels 1 et 2. Pour Label-composition 1, la condition est $v_9 > v_{11}$ et $v_{10} > v_8$ qui est fausse donc la composition n'est pas valide. Pour Label-composition 2, la condition est $v_9 < v_{11}$ et $v_{10} < v_8$ qui est vraie donc la composition est valide et l'anomalie Pic Négatif est reconnue en v_{10}.

Enfin, nous avons mis en œuvre un algorithme capable de parcourir une liste labellisée Y_L et vérifier, à partir de chaque point, quelles compositions de labels s'appliquent pour identifier les anomalies (et les points correspondants).

4. Expérimentations

4.1. Description d'étude de cas

Le domaine d'application traité dans cet article est le réseau de capteurs du Service de gestion et d'exploitation (SGE) du campus de Rangueil rattaché au rectorat de Toulouse. Ce service exploite et entretient le réseau de distribution à partir des données liées aux différentes installations. Plus de 600 capteurs de différents types de fluides (calories, eau, air comprimé, électricité et gaz), disséminés dans plusieurs bâtiments, sont gérés par les systèmes de supervision SGE. Pour cet article, nous nous sommes concentrés sur les données de calories et nous avons traité 25 capteurs. Ainsi, nous

analysons les mesures de calories collectées chaque jour pendant plus de trois ans par 25 capteurs déployés dans différents bâtiments (1453 données par capteur soit 36325 points de données au total). Les mesures de ces capteurs sont rassemblées à une fréquence régulière et représentent les *index* (lectures de capteurs) et sont ensuite utilisées pour mesurer les *quantités d'énergie consommées* (par différences de valeur d'index successives). Nous avons pu identifier les types d'anomalies et les points concernés présents dans les données de capteurs de calorie grâce aux connaissances acquises par les experts du SGE et à travers une inspection manuelle d'un ensemble de capteurs de même type que les capteurs étudiés.

Les défauts présentés figure 1 sont extraits de ces mêmes ensembles de données. L'anomalie prédominante est constituée par les valeurs constantes, près de 8578 observations sur toutes les données, suite à un arrêt de capteurs. Nous avons également trouvé parmi ces valeurs plusieurs constantes avec un décalage. Généralement, une constante avec un décalage de niveau commence par un pic positif ou négatif. Ensuite, il existe beaucoup de changements anormaux tels que des pics positifs ou négatifs, près de 380. Enfin, il y a huit changements de niveau dus au changement de capteur. Afin de détecter les points remarquables et les anomalies, nous avons construit, avec l'aide des experts, 14 patterns et 12 compositions de labels pour détecter les anomalies décrites ci-dessus.

4.2. Expérimentation sur des données réelles (séries croissantes et séries variables)

Dans cette partie, nous explorons les différentes méthodes de détection d'anomalies décrites dans le tableau 1. Notre motivation à envisager ces méthodes est d'élargir le champ d'analyse afin de tester leur efficacité dans la détection d'anomalies cherchées dans cet article et de comparer les résultats avec notre approche. Comme indiqué ci-dessus, ces techniques se sont révélées efficaces pour détecter les anomalies dans les données du capteur. Cependant, comme nous le verrons à travers les résultats des expériences, aucune de ces méthodes n'est efficace pour détecter tous les types d'anomalies de manière simultanée dans les déploiements réels. Nous présentons une évaluation des méthodes suivantes : Règle Courte (noté SR), Règle Constante (noté CR), LOF, ARIMA, S-H-ESD et Change Point (noté LS). Nous avons appliqué ces méthodes par catégorie d'anomalies comme indiqué dans le tableau 1. Afin d'évaluer leurs performances, nous utilisons le nombre de vrais positifs (vraies anomalies détectées), le nombre de faux positifs (fausses anomalies détectées) et le nombre de faux négatifs (vraies anomalies non détectées) en tant que métriques d'évaluation.

Nous présentons les résultats de LOF dans un graphique séparé pour plus de lisibilité (figure 6B). Ces méthodes ne sont pas entièrement automatisées et nous devons donc sélectionner les paramètres tels que le seuil pour la Règle Courte, le nombre de voisins et le seuil pour évaluer le score du degré d'anomalie pour LOF ou le type de modèle pour ARIMA, etc. Pour LOF, nous avons fait varier le choix de K (nombre de voisins) dans une plage de 30 à 10 afin d'évaluer l'influence de ce paramètre sur le résultat de la détection (cf. haut de la figure 6B) et nous avons déterminé un seuil

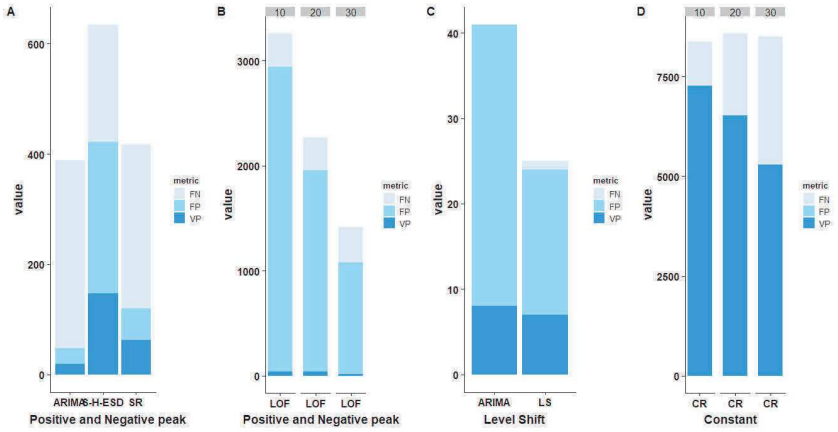

Figure 6. Evaluation des méthodes de détection d'anomalies sur les données d'index.

= 1,5 qui correspond à une distribution standard. Pour la Règle Courte, nous avons choisi un seuil en utilisant la méthode basée sur l'histogramme décrite dans la section 2. Enfin, pour la règle constante, nous avons fait varier le choix de la taille de la fenêtre coulissante dans une plage de 30 à 10 comme le montre la figure 6D (haut de la figure).

En se basant sur les résultats présentés figures 6A, 6B, 6C, et 6D, nous pouvons faire les observations suivantes : LOF est la méthode qui génère le plus de faux positifs tandis ARIMA génère le plus de faux négatifs. La méthode S-H-ESD est celle qui permet de détecter le plus de vrais positifs par rapport à LOF, ARIMA et RC, par contre elle génère beaucoup de faux positifs et de faux négatifs. La Règle Courte détecte moins d'anomalies en comparaison de S-H-ESD. Cependant, il en résulte moins de faux positifs que les autres méthodes. Également, nous pouvons dire que le nombre de voisins égal à 20 est le choix le plus approprié pour détecter le plus grand nombre d'anomalies pour l'algorithme LOF. Cependant, pour la Règle de Constante, il est important d'utiliser une taille de fenêtre suffisamment petite pour gérer des données contenant plusieurs anomalies constantes (ici 10).

Le tableau 2 présente les résultats de ces méthodes en fonction de la précision, du rappel et de la F-mesure sur les données d'index (série croissante) et sur les données de consommation (série variable).

Concernant les données d'index, en se basant sur ce tableau et la figure 6 nous déduisons que : (i) l'efficacité de la Règle Constante ou de la méthode LOF dépend fortement du choix de la fenêtre glissante ou du nombre de voisins ; (ii) la méthode Change Point fonctionne bien lorsqu'il y a réellement un changement de niveau dans la série temporelle, mais cependant, en cas d'absence d'anomalie, sa précision est faible ; et (iii) entre la Règle Courte, ARIMA et S-H-ESD, la Règle Courte est la plus précise et ARIMA est la moins efficace pour détecter un changement anormal. En comparant avec ces méthodes, notre algorithme CoRP arrive à détecter de multiples

Tableau 2. Comparaison des méthodes de détection d'anomalies sur les données d'index et de consommation.

Données	Séries croissantes (Index)			Séries variables (Consommation)		
	Pre	Rec	$F-m$	Pre	Rec	$F-m$
SR	0.52	0.17	0.32	0.66	0.63	0.64
CR	1	0.80	0.88	1	0.72	0.83
LOF	0.022	0.12	0.022	0.39	0.78	0.52
$S-H-ESD$	0.34	0.40	0.36	0.41	0.80	0.54
$ARIMA$	0.30	0.07	0.11	0.66	0.25	0.36
LS	0.29	0.87	0.43	X	X	X
$CoRP$	1	1	1	1	0.98	0.98

types d'anomalies simultanément avec une meilleure précision et un meilleur rappel. En effet, CoRP fonctionne très bien sur les données d'index représentant notre étude de cas.

Pour évaluer davantage notre algorithme et le confronter aux méthodes de détection d'anomalies, nous avons utilisé les données de consommation du SGE. Nous avons donc pris les mesures provenant des 25 capteurs. Les données de consommation sont des données saisonnières et leur évolution quotidienne, contrairement aux données d'index, est variable. Nous avons inspecté manuellement un ensemble de données de même type pour comprendre leurs variations et créer les patterns de détection des points remarquables qui peuvent exister puis les compositions de labels pour détecter les anomalies. Les anomalies observées dans ces données sont les suivantes : pics positifs et négatifs, anomalies constantes, anomalies constantes commençant et se terminant par un décalage important. Ainsi, toujours avec l'aide des experts, nous avons créé 9 patterns afin de détecter les points remarquables et 5 compositions de labels pour détecter les anomalies.

Nous rapportons les résultats des algorithmes sur les données de consommation dans le tableau 2 (séries variables). Comme les données ne sont pas stationnaires, nous n'avons pas appliqué l'algorithme Change Point parce qu'il n'existe pas de changement de niveau dans ces données à détecter. Ce tableau montre que les algorithmes de la littérature sont beaucoup plus efficaces sur les données de consommation en comparant avec les résultats sur les données d'index. Mais même sur ce type de données, notre approche a obtenu le meilleur résultat de F-mesure en comparant avec les autres algorithmes. En effet, CoRP a détecté le plus d'anomalies avec le moins d'erreurs possibles, avec une précision égale à 1 et un rappel égal à 0,98. Notons que, les résultats de la méthode à base de règles (SR, CR) et de la méthode ARIMA ont une meilleure précision par rapport à LOF et SH-ESD. Enfin, SH-ESD est la méthode la plus proche du meilleur résultat en termes de rappel avec une valeur égale à 0,80. Toutefois, il faut noter que ces algorithmes n'arrivent pas détecter simultanément tous les types d'anomalies observées dans les déploiements réels, ce qui signifie que chaque algorithme est efficace dans un type spécifique. La particularité de notre méthode est que

nous pouvons définir les patterns en fonction de nos besoins afin de détecter avec une grande précision et efficacité de multiples anomalies simultanées.

4.3. Expérimentation sur des données de la littérature (séries variables)

Afin d'évaluer l'algorithme dans un autre contexte, nous avons utilisé les ensembles de données proposés dans le package d'implémentation de la méthode ARIMA (Lacalle, 2016). Parmi ces données, nous avons exploré les données de HIPC (Harmonised Indices of Consumer Prices). Ces ensembles de données représentent les indices harmonisés des prix à la consommation dans la zone euro. Nous avons également exploré les données IPI (Industrial Production Indices). Ces données représentent les indices de la production industrielle dans le secteur manufacturier des pays de l'Union monétaire européenne (Lacalle, 2016). Chacun de ces ensembles de données contient plusieurs séries temporelles qui présentent des données mensuelles de 1995 à 2013. Un premier sous-ensemble comportant deux séries temporelles de ces deux ensembles de données a permis de mener les expérimentations. Chacune de ces séries contient 229 mesures avec 5 anomalies en HIPC et 4 anomalies en IPI. Ces anomalies sont variées : AO (Additive Outlier), TC (Temporary Changes) ou LS (Level Shift). Un deuxième sous-ensemble de séries nous a permis de spécifier les patterns en définissant un pattern différent par type d'anomalies pour labelliser les points remarquables dans la série temporelle (3 patterns). Ensuite, nous avons procédé à une composition de ces labels pour détecter les anomalies (4 compostions de labels).

Tableau 3. Comparaison de méthodes de détection d'anomalies sur des données de benchmark.

Datasets	HIPC		IPI	
Evaluation / Algorithm	Precision	Recall	Precision	Recall
CoRP	1	0.80	0.75	0.75
ARIMA	1	1	1	1
LOF	0.11	0.20	0	0
S-H-ESD	0.20	0.20	0.33	0.25
RC	0	0	0	0
LS	0	0	0	0

Le tableau 3 est une comparaison entre les algorithmes de la littérature et notre algorithme sur les données proposées dans le package ARIMA. Nous n'avons pas testé la Règle Constante dans les ensembles de données HIPC et IPI car les anomalies observées dans ces données ne contiennent pas ce type d'anomalie. Nous avons par conséquent appliqué CoRP, ARIMA, LOF avec un nombre de voisins égal à 20, S-H-ESD, Change Point et la Règle Courte sur ces données. L'algorithme basé sur la Règle Courte et Change Point sont les moins précis parmi ces algorithmes, tandis que notre algorithme est le meilleur parmi eux et peut détecter la majorité des anomalies observées avec peu d'erreurs.

4.4. Temps de calcul

Dans cette partie, nous nous abordons le temps de calcul requis pour les différentes méthodes de la littérature et notre algorithme. Les expériences sont effectuées sur une machine exécutant Windows 10 Professional, un processeur Intel (Core) i5 et 16 Go de RAM. Nous avons utilisé la distribution open source Python 3.7 Anaconda pour développer notre algorithme et R 3.5 pour explorer les algorithmes de la littérature. Nous avons calculé le temps d'exécution sur les 25 séries de données d'index pour chaque algorithme évalué afin de le comparer au temps d'exécution de notre algorithme. Le classement des algorithmes en fonction de leurs performances d'exécution est : la méthode à base de règles et la méthode S-H-ESD sont les plus rapides avec un temps d'exécution de 0.5s. Ensuite, la méthode LOF prend un temps d'exécution égal à 2.5s. Notre algorithme prend une durée d'exécution de 5,40 secondes et enfin ARIMA demande 7,60 secondes.

5. Conclusion

La détection d'anomalies dans les applications de supervision est très importante notamment dans le domaine des réseaux de capteurs. Cet article présente l'approche CoRP basée sur des patterns appliqués aux séries temporelles uni-variées de données de capteurs. Notre méthode est composée de deux étapes : elle marque (labels) tous les points remarquables présents dans la série temporelle sur la base de patterns de détection, puis, elle identifie précisément les multiples anomalies présentes par compositions de labels. Cette approche nécessite l'expertise du domaine d'application pour pouvoir définir efficacement les patterns et les compositions de labels. A l'inverse, les méthodes de la littérature, bien que moins efficaces, ne demandent pas autant d'expertise métier pour être appliquées.

Notre expérimentation est basée sur un contexte réel : les données de capteurs du SGE (service de gestion et d'exploitation du campus de Rangueil à Toulouse). L'évaluation de cette méthode est illustrée en utilisant tout d'abord les données d'index et de consommation des capteurs de calories exploités par le SGE et, en second lieu, en utilisant des jeux de données issus de l'état de l'art. Nous comparons notre algorithme à cinq méthodes appartenant à différentes techniques de détection d'anomalies. Sur la base des critères d'évaluation précision, rappel, f-mesure, nous montrons que notre algorithme est le plus efficace pour détecter de manière simultanée différents types d'anomalies observées lors de déploiements réels en minimisant les fausses détections. Plusieurs extensions à notre approche sont envisagées : (i) afin de moins nécessiter l'intervention d'experts du domaine, utiliser des méthodes d'apprentissage pour construire automatiquement les patterns et/ou les compositions de labels ; (ii) appliquer notre algorithme sur des flux de données réels pour fournir les alarmes en identifiant les anomalies le plus tôt possible.

Remerciements

Ce travail de thèse est financé le service de gestion et d'exploitation (SGE) de Rangueil rattaché au rectorat de Toulouse. Les auteurs remercient le SGE pour avoir fourni l'accès aux données réelles de capteurs. Ils remercient également les experts qui ont aidé à la compréhension de ces données et l'identification des anomalies observées lors de l'exploitation.

Bibliographie

Aminikhanghahi S., Cook D. J. (2017). A survey of methods for time series change point detection. *Knowledge and information systems*, vol. 51, n° 2, p. 339–367.

Basseville M., Nikiforov I. V. *et al*. (1993). *Detection of abrupt changes: theory and application* (vol. 104). Prentice Hall Englewood Cliffs.

Breunig M. M., Kriegel H.-P., Ng R. T., Sander J. (2000). Lof: identifying density-based local outliers. In *ACM SIGMOD record*, vol. 29, p. 93–104.

Chandola V., Banerjee A., Kumar V. (2009). Anomaly detection: A survey. *ACM computing surveys (CSUR)*, vol. 41, n° 3, p. 15.

Chen C., Liu L.-M. (1993). Joint estimation of model parameters and outlier effects in time series. *Journal of the American Statistical Association*, vol. 88, n° 421, p. 284–297.

Cleveland R. B., Cleveland W. S., McRae J. E., Terpenning I. (1990). Stl: A seasonal-trend decomposition. *Journal of Official Statistics*, vol. 6, n° 1, p. 3–73.

Hochenbaum J., Vallis O. S., Kejariwal A. (2017). Automatic anomaly detection in the cloud via statistical learning. *arXiv preprint arXiv:1704.07706*.

Hodge V., Austin J. (2004). A survey of outlier detection methodologies. *Artificial intelligence review*, vol. 22, n° 2, p. 85–126.

Lacalle J. López-de. (2016). tsoutliers R package for detection of outliers in time series. *CRAN, R Package*.

Ramanathan N., Balzano L., Burt M., Estrin D., Harmon T., Harvey C. *et al*. (2006). Rapid deployment with confidence: Calibration and fault detection in environmental sensor networks.

Rosner B. (1983). Percentage points for a generalized esd many-outlier procedure. *Technometrics*, vol. 25, n° 2, p. 165–172.

Sharma A. B., Golubchik L., Govindan R. (2010). Sensor faults: Detection methods and prevalence in real-world datasets. *ACM Transactions on Sensor Networks (TOSN)*, vol. 6, n° 3, p. 23.

Sreevidya S. *et al*. (2014). A survey on outlier detection methods. *International Journal of Computer Science and Information Technologies (IJCSIT)*, vol. 5, n° 6.

Upadhyaya S., Singh K. (2012). Nearest neighbour based outlier detection techniques. *International Journal of Computer Trends and Technology*, vol. 3, n° 2, p. 299–303.

Yao Y., Sharma A., Golubchik L., Govindan R. (2010). Online anomaly detection for sensor systems: A simple and efficient approach. *Performance Evaluation*, vol. 67, n° 11, p. 1059–1075.