SCIS 2006 The 2006 Symposium on Cryptography and Information Security
Hiroshima, Japan, Jan. 17-20, 2006
The Institute of Electronics, Information and Communication Engineers

変化点検出エンジンを利用したインシデント検知システムの構築

Development of Incident Detection System Based on Change Point Detection

竹内 純一*† 佐藤 靖士‡ 力武 健次* 中尾 康二*§
Jun-ichi Takeuchi Yasushi Sato Kenji Rikitake Koji Nakao

あらまし 情報通信研究機構 (NICT) セキュリティ高度化グループは，インターネット上で起こるセキュリティインシデントを分析する「インシデント分析システム」の構築を行っている．本システムの重要な機能の 1 つに，リアルタイムインシデント検知がある．これを実現する試みとして，時系列データの急激な増加などを高速高精度に検出する変化点検出エンジン ChangeFinder を応用した検知システムを構築した．本論文ではシステムの概要と実際のデータを用いた検出例について報告する．

キーワード セキュリティインシデント，異常検出，変化点検出

1 はじめに

情報通信研究機構 (NICT) セキュリティ高度化グループでは，インターネット上で起こるセキュリティインシデントを分析する「インシデント分析システム」の構築を行っている [10]．このシステムの目的は，ネットワークのリスク状況を把握し，危険レベル及びネットワークに対する影響度を導出し，日本の情報通信基盤の安全性確保に貢献することにある．重要な要件の 1 つに，各種ログデータからのインシデント候補のリアルタイム検知がある．本稿では，この機能を担うモジュールについて述べる．検知機能には複数の異なる手法を適用するが，本稿で述べるのは，時系列データの変化点をリアルタイムに検出する変化点検出エンジンを応用したシステムである．

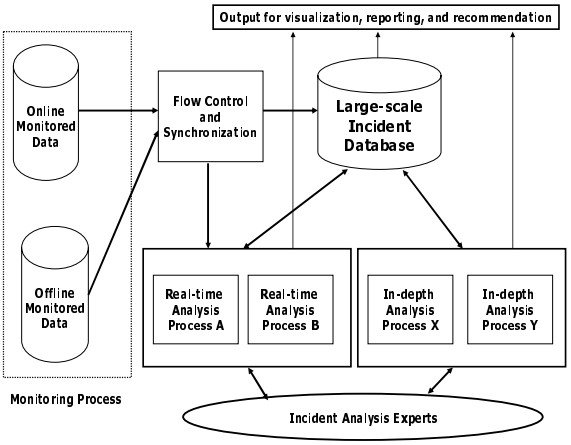

図 1: インシデント分析システム

2 インシデント分析システム

NICT のインシデント分析システム (図 1) の全貌については [10] で述べているが，以下に概略を示す．

インシデント分析システムは以下の要件を満たすことを目標としている：1) 日本全域にまたがる ISP やエッジユーザなどからのイベントを分析対象とすること，2) イベントデータの収集から，インシデント候補のピックアップまでのタイムラグを 2 分以内とする**リアルタイム性**を確保すること，3) 上記実時間分析と並行し，詳細分析を行うこと，4) インシデントの検知のみでなく，検知したインシデントに対する有効な対策を導出すること，である．

上記 1)-3) に対応した機能をそれぞれ，イベント収集装置，実時間分析，詳細分析と呼ぶ．このうち，実時間分析を目的として「変化点分析」,「稀率分析」[7],「振る舞い分析」[9] 「Bot 検出とマルウェア解析」の四つのモジュールを備える．これらに要求される機能は，日本全域に仕掛けられたイベント収集装置が集めるログデータをリアルタイムに分析し，インシデントの候補となる

* 独立行政法人 情報通信研究機構 セキュリティ高度化グループ
〒 184-8795 東京都小金井市貫井北町 4-2-1
Security Advancement Group, National Institute of Information and Communication Technology / 4-2-1 Nukui-Kitamachi, Koganei City, Tokyo 184-8795, Japan
† NEC インターネットシステム研究所
Internet Systems Research Labs., NEC Corp.
‡ NEC システム基盤ソフトウェア開発本部
System Platform Software Development Division, NEC Corp.
§ KDDI 株式会社 情報セキュリティ部
Information Security Department, KDDI Corp.

異常を逸早く発見することであり，多量のログデータのリアルタイム処理というデータマイニングの課題として捉えることが出来る．この考えに基づき，変化点分析システムを，汎用の変化点検出エンジンとして開発された ChangeFinder を主要コンポーネントとして構築している．

ChangeFinder の機能は，数値データの時系列を入力とし，各時点に対応して変化点らしさを示す「変化点スコア」を出力するものである．従って，イベント収集装置が集める各種ログを，数値的な統計量に集計する前処理を行う．例えば，特定のポートへの単位時間あたりのアクセス数の時系列や，振る舞い分析によって得られる特定のアクセスパターンの発生頻度，さらにこれら統計量を組み合わせた合成変数などを入力として用いることが出来る．こうした人間が観察するのが不可能なほど多数の統計量を並列で処理し，インシデントの候補となる変化の発生を知らせることが，変化点分析システムの目的である．

以下，第 3 節で変化点検出問題と ChangeFinder について簡単に解説し，第 4 節で，現在構築している変化点分析システムの概要を述べる．その後，第 5 節で，セキュリティインシデントの検出例を示す．

3　変化点検出

本節で述べる変化点とは，基本的に定常な時系列データの情報源の性質が突然変化する時点のことを指す．変化点検出とは，そうした変化点を特定する問題である．このような変化点検出の最も基本的な方式は，Guralnik and Srivastava [2] の仕事に見られるように，データ列に AR モデル (付録 A 参照) や多項式回帰モデルなどの時系列モデルをあてはめていき，変化点の候補となる点の前後で別々に時系列モデルを当てはめた場合が，そうしない場合に比べて当てはめ誤差を有意に少なくできるかどうかを検定し，YES ならば，その候補点を変化点とみなすという方式である．

こうした方式は，一般に計算時間がかかり，インシデント検知のように一刻を争う問題には適さない．これに対し，よりシンプルでオンライン処理に向いた変化点検出アルゴリズムを，Yamanishi and Takeuchi [5, 4, 11] が提案している．これは，時系列データを 2 段階に渡って学習し，各時点の変化点スコアをオンラインで算出するものである (図 2)．本システムでは，学習には AR モデルを用いるが，本来はこれに限定しない．また，その学習手法には，過去の情報を少しずつ忘却しながら学習する SDAR(Sequential Discounting AR estimating) アルゴリズムを用いる．これについても付録 A を参照されたい．このアルゴリズム (以下 ChangeFinder と書く) の原理を示す．

第一段階学習：各時刻 t において，AR モデルを SDAR アルゴリズムによって学習する．学習した確率密度関数を $p_t(x)$ とする[1]．各時刻 t についてデータ x_t の外れ値スコアを対数損失 $-\ln p_{t-1}(x_t)$ として計算する．

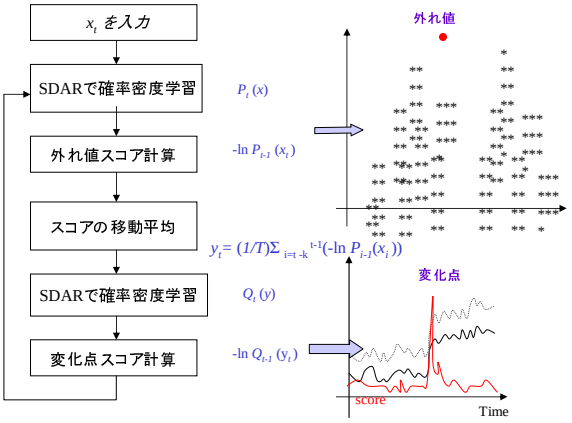

図 2: ChangeFinder の処理の流れ

平滑化：一定サイズ T のウインドウ内のデータの外れ値スコアの平均 $y_t = (1/T)(\sum_{i=t-T}^{t-1}(-\ln p_i(x_i))$ を計算し，ウインドウをスライドすることによって移動平均スコアの時系列 $y_t : t = 1, 2, ..$ を構成する．

第二段階学習：この時系列に対して再度 AR モデルをあてはめ，これを学習して，その系列を $q_t(x)$ とする．各時点 t において対数損失 $-\ln q_{t-1}(y_t)$ を時刻 t の変化点スコアとして計算する．変化点スコアが大きいほど，t が変化点である度合いが高い．

ChangeFinder の鍵は，第一段階学習では時系列中の外れ値しか検出できないところを，外れ値スコアの平滑化を通じて，本質的なモデルの変動を検出しているところにある．計算量としては，データ数 n に関しては，統計的検定に基づく方式が $O(n^2)$ であるのに対して，ChangeFinder の計算量は $O(n)$ で済むので，後者の方がはるかに効率的である．

図 3 に検出精度の定量評価の例を示す．ここでは，変化点スコアに対して閾値を設けて警報を上げる設定を行った．グラフの横軸は誤報率であり，閾値によってコントロールする．縦軸は平均 benefit で，変化点をいかに正確に検出したかを示す指標である．定義を付録 B に示した．CF は ChangeFinder，GS は Guralnik and Srivastava の手法で線形回帰を用いたもの，SC は AR モデルによる統計的検定に基づく手法を示している．用いたデータは 10,000 レコードからなり，1000 レコード毎に平均値が 5 だけ不連続に変化する変化点を配置した．変化点と変化点の間は，分散パラメータが 1 の AR モデルに従う定常時系列である．これによると，CF は誤警報率が 0.1 以下の領域を除いては最も良い benefit を達

[1] AR モデルは定常性を仮定する統計的モデルであるが，SDAR アルゴリズムの忘却型学習によって非定常性も扱うことができる．具体的アルゴリズムについては，付録および [4] を参照されたい．

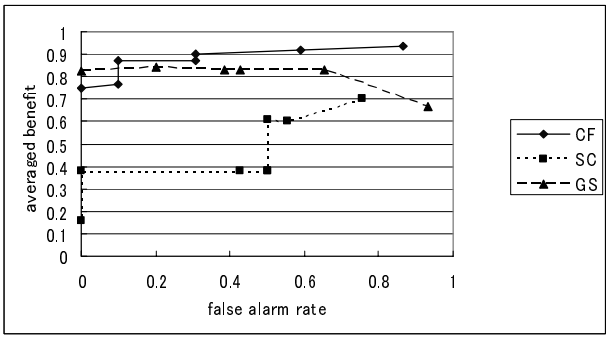

図 3: ChangeFinder の検出精度

成していることが分かる．

また，ChangeFinder は上記のように平均値が変化する視覚的に分かりやすい変化だけでなく，AR モデルのパラメータ (AR 係数や分散) の変化も原理的には検出できる．実際，[4] では分散が突然変化する場合の評価も行い，CF のみが有効に検出できることを確認している．この特長は，目視では見分けにくいインシデントの予兆を捉えるのに役立つ可能性がある．

4 変化点検出によるインシデント検知システム

本節では，3 節で説明した ChangeFinder を，図 1 における Real-time Analysis Process (実時間解析プロセス) の 1 つとして組み込み，実装したシステムについて解説する．

以下に，変化点検出によるインシデント検出システムの概要図を示す．

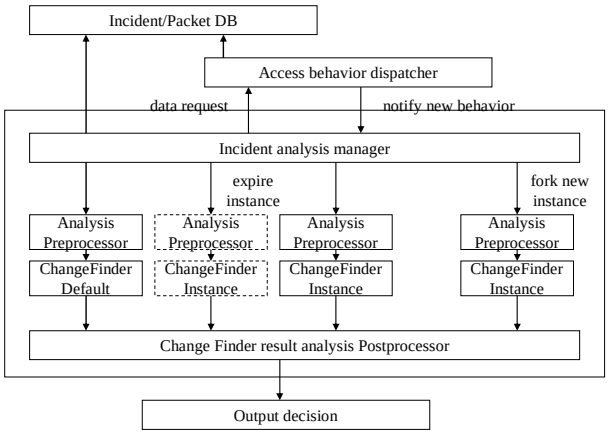

図 4: 変化点検出によるインシデント検出システム

本検出システムは，以下のモジュールによって構成される．

- インシデント分析マネージャ(Incident analysys manager)
- 分析プリプロセッサ (Analysys Preprocessor)
- ChangeFinder
- 分析ポストプロセッサ (ChangeFinder result analysis Postprocessor)

なお，分析プリプロセッサと ChangeFinder の 1 つの組み合わせを，特定パターン分析を行う**分析ライン**と呼ぶ．本システムでは規定の分析パターンで変化点分析を行うデフォルト分析ラインと，新規の分析パターンを動的に追加して分析するための派生分析ラインを並列で実行する事により，既知のインシデントと未知のインシデントの両方を同時に追跡可能なシステムとなっている．

以下に，上述の各モジュールについて説明する．

インシデント分析マネージャ

分析対象となるデータを上位のシステムから受信し，あるいは上位システムに要求してデータを取得すると共に，規定の設定に従い分析ラインにデータを引き渡す機能を持つ．更に，本分析マネージャは，実時間分析モジュールの 1 つとなる振る舞い分析モジュールと連携した解析も行う．振る舞い分析では，アクセス状況を複数の振る舞いパターンに分類する機能を持つ (図 5 参照)．これらのパターンの発生頻度を時系列として変化点分析

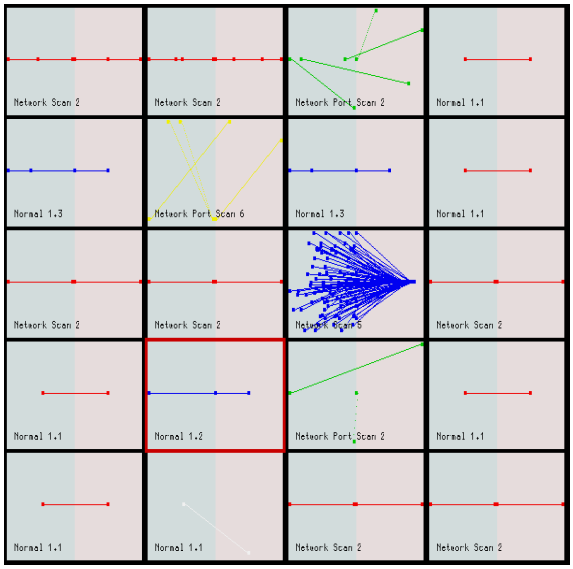

図 5: 振る舞い分析によるパターン分類例

にかけることが出来る．特に，動的に検出された新規の振る舞いパターンの通知を受け取り，対応する分析ラインを新規に起動して，該当パターンに対する変化点検出を開始する機能をもつ．このようにして動的に起動された分析ラインは，一定時間，あるいは設定により規定された条件に従って，新規に通知された振る舞いパターンについて監視し，分析不要と判断した場合には自動的に分析を停止して，分析ラインを削除する．

分析プリプロセッサ

分析対象のデータを加工し，ChangeFinder を起動する機能を持つ．この機能では分析対象となるデータについて，以下の処理を行う．

- パラメータ抽出
- 基礎統計量作成 (時系列情報生成)
- データグルーピング

パラメータ抽出 上位システムより受信したデータから，ChangeFinder で監視すべきデータを抽出し，複数のデータ系列を生成する．例えば検知対象となるパケットの情報には，送受信アドレスや，ポート番号，プロトコルタイプといった様々なパラメータが含まれているが，これらの情報を一旦ソースポート単位やソースアドレス単位等の組み合わせに分類する．

基礎統計量作成 パラメータ抽出によって複数のデータ系列化された情報から，変化点検出を行う為の一定時間毎の統計量を作成する．

データグルーピング 前述のパラメータ抽出と基礎統計量作成の処理によって，元のデータから複数の時系列の基礎統計量が作成されるが，既知のインシデントの幾つかのケースについては，その数値に相関がある事が事前に判明している場合がある．例えばある不正アクセスの形態では，まず Port 138 をアクセスし，次に Port 445 をアクセスする，といったケースである．このようなケースでは，両方の統計量を組み合わせて変化点を検出するほうが有効であるため，本処理で関連する複数パラメータを組み合わせ，ChangeFinder に対してまとめて受け渡す処理を行う．

ChangeFinder

第 3 節で解説した ChangeFinder を起動し，分析プリプロセッサによって生成されたデータを分析する．分析結果は，後段の分析ポストプロセッサに引き渡される．

分析ポストプロセッサ

ChangeFinder が求めた変化点スコアに対して一定の基準を設定し，インシデントの判定を行う．

本システムは，ネットワークを流れるパケットの情報を元にその変化点を検出しているため，その中にはワームの発生等によるパケットパターンの増加ではなく，単に特定のユーザ間の通信が増加したケースでも変化点スコアが上昇する可能性がある．そのため，インシデントの出力として変化点スコアをそのまま利用するのではなく，閾値による判定や，各スコアの組み合わせなどにより，結果を総合的に判断してインシデント検知の警報を出力する．

このように本システムでは，新規のパケットパターンが発生した場合に，そのパターンに該当するアクセス数が一定の期間内に全く変化しない，あるいは急激に増加する等の変化点を検出することによって，未知のインシデントの発生を早期に検出する事が可能なシステムとなっている．

5 検出例

前節で説明した変化点分析システムによるインシデントの検出例を示す．インシデント分析システム内に蓄積してある過去の状況を再現する機能を用い，2004 年 4 月〜5 月の Sasser ウイルス発生時のデータを変化点分析システムに入力した．図 6 は，このうち TCP ポート 445 へのアクセス数を処理した例である．Sasser-A が発生した 4 月 30 日の直前に高い変化点スコアが出ていることが分かる (図 6 中の 4/25 付近の高いスコアは学習が不十分であるために生じるノイズ)．

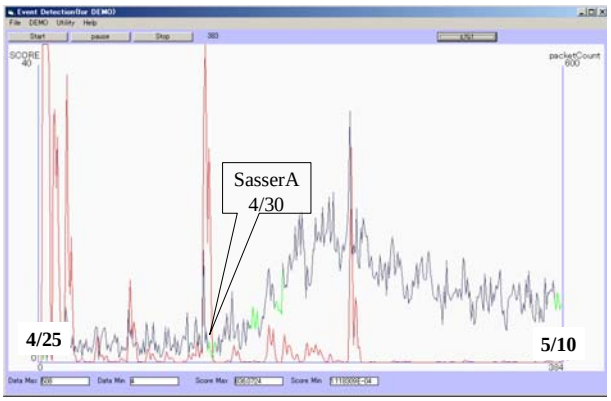

図 6: Sasser ウイルスを検出

6 むすび

インシデント分析システムの一部として構築している変化点分析モジュールについてその数学的根拠と実装手法，インシデント検知システムへの応用と実際の検出例について述べた．今後はこうしたシステムでは完全に避けるのが難しい誤警報を，他の実時間分析モジュールの出力と組み合わせるなどして出来るだけ減らす方法を模索したい．

謝辞 本システム構築についてご支援を頂いている横河電機の馬場俊輔氏，鈴木和也氏，実装を担当して頂いている NEC 情報システムズの白戸幸正氏，村田康裕氏，Sasser ウイルスについてご教示頂いた NEC の清水雅子氏，本システムについて議論して頂いている NICT セキュリティ高度化グループと分析チーム各位，そして筆

者の一人 (竹内) が日頃多くの助言と示唆を頂く NEC の山西健司氏に感謝いたします.

参考文献

[1] T. Fawcett and F. Provost, "Activity monitoring: noticing interesting changes in behavior," *Proc. of KDD-99*, pp:53–62, 1999.

[2] V. Guralnik and J. Srivastava, "Event detection from time series data," *Proc. KDD-99*, pp:33–42, 1999.

[3] G. Kitagawa and W. Gersch, *Smoothness Priors Analysis of Time Series*, Lecture Notes in Statistics, 116, Springer-Verlag, 1996.

[4] J. Takeuchi and K. Yamanishi, "A Unifying Framework for Detecting Outliers and Change Points from Time Series," *IEEE transactions on Knowledge and Data Engineering,* in print.

[5] K. Yamanishi and J. Takeuchi, "A Unifying Approach to Detecting Outliers and Change-Points from Nonstationary Data," *Proc. of KDD2002*, 2002.

[6] B.-K. Yi., ND Sidiropoulos, T. Johnson, HV Jagadish, C. Faloutsos, A. Biliris, "Online Data Mining for Co-Evolving Time Sequences," *Proc. of the 16th International Conference of Data Engineering,* 2000.

[7] 竹森敬祐，三宅優，田中俊昭，"ネットワーク間プロファイル比較による攻撃異常検知," **情報処理学会研究報告** 2004-DPS-122, Vol. 2005, No. 33, pp. 87-92，2005 年 3 月.

[8] 尾崎統，北川源四郎 編，**時系列解析の方法**，朝倉書店，1998.

[9] 鈴木和也，馬場 俊輔，田中 貴志，金山 卓矢，"トラフィック監視による新出ワームの検出システム," 信学技報，vol. 104, no. 377, IA2004-14, pp. 7-12, 2004.

[10] 中尾康二，力武健次，竹内純一，大河内一弥，吉岡克成，衛藤将史，守山栄松，松本文子，"インターネットにおける実時間イベント分析の研究開発," *SCIS2006* 採録予定, 2006.

[11] 山西健司，竹内純一，丸山祐子，"統計的異常検出 3 手法," **情報処理**，Vol. 46, No. 1, pp. 34-40, 2005.

付録

A　AR モデルとその学習

ChangeFinder と，比較対照アルゴリズムの SC では AR モデルを用いている．これは標準的な時系列モデルの一つである ([3],[8]).

今，期待値が 0 であるような定常時系列 $\{z_t : t = 1, 2, \cdots\}$ が k 次の AR モデルに従って発生すると仮定すると，

$$z_t = \sum_{i=1}^{k} A_i z_{t-i} + \varepsilon,$$

で表される．ただし，データ z_t は n 次元のベクトル，$A(i)(i = 1, ..., k)$ は n 元正方行列，ε は期待値 0，共分散行列 Σ のガウス分布 $\mathcal{N}(0, \Sigma)$ に従うノイズ項であるとする．実際に観測されるデータを $x_t = z_t + \mu$ で表す．すなわち，期待値が μ であるとする．$x_{t-k}^{t-1} = (x_{t-1} \cdots x_{t-k})$ とするとき，x_t の確率密度関数は

$$p(x_t | x_{t-k}^{t-1} : \theta) = \frac{1}{(2\pi)^{k/2} |\Sigma|^{1/2}} \exp\left(-\frac{\xi^T \Sigma^{-1} \xi}{2}\right) \quad (1)$$

で与えられる．ただし，$\xi = x_t - (\sum_{i=1}^{k} A(i) z_{t-i} + \mu)$ であり，$\theta = (A_1, \cdots, A_k, \mu, \Sigma)$ と置いた.

まず，AR モデルに関する通常の最尤推定アルゴリズムを復習する ([8] 参照)．以下の量を定義する．

$$\hat{\mu} = \frac{1}{t-k} \sum_{i=k+1}^{t} x_i \quad (2)$$

$$C_j = \frac{1}{t-k} \sum_{i=k+1}^{t} (x_i - \hat{\mu})(x_{i-j} - \hat{\mu})^T \quad (3)$$

(2) は μ の推定値を表し，(3) は $x_1, \cdots, x_t$ の相関関数の推定値を表す．さらに $A(i)$ の推定値は，以下の $\bar{A}_i$ を未知数とする連立方程式を解くことで得られる．

$$C_j = \sum_{i=1}^{k} \bar{A}(i) C_{j-i} \quad (j = 1, \cdots, k). \quad (4)$$

(4) の解を $\hat{A}(i)$ とおくと，Σ の推定値は

$$\hat{\Sigma} = C_0 - \sum_{i=1}^{k} \hat{A}(i) C_i \quad (5)$$

によって求められる．この手続きでは，情報源が定常であると仮定されていること，また，いわゆるバッチ学習方式となっていることに注意されたい.

この方式を改良し，**SDAR**(**sequentially discounting AR model estimating**) アルゴリズムと名づける逐次型学習方式を導入する．ポイントは以下の二点である.

1) **逐次学習**; 新たなデータを一つ読み込むごとにパラメータの推定値を更新する.

2) **忘却機能**; i 時点前のデータの影響が $(1-r)^i$ 倍に減少するようにパラメータの推定値を更新する．これによって，非定常な情報源に対応できる．アルゴリズムのパラメータ r を**忘却パラメータ**と呼ぶ．r の解釈としては，$1/r$ 個程度の過去データの情報を蓄積すると考えればよい．例えば $r = 0.01$ とすれば過去 100 個ほどのデータの傾向を持った AR モデルが獲得される．なお，類似のアルゴリズムが [6] で提案されている．

SDAR アルゴリズムを図 7 に示す．

SDAR アルゴリズム $(0 < r < 1$:所与$)$

Step 1. **初期化**

Set $\hat{\mu}, C_j, \hat{A}(j)\ (j = 1, \cdots, k), \hat{\Sigma}$.

Step 2. **パラメータ更新**

For $t = 1, 2, \cdots,$

x_t を読み込む:

$$\begin{aligned} \hat{\mu} &:= (1-r)\hat{\mu} + r x_t \\ C_j &:= (1-r)C_j + r(x_t - \hat{\mu})(x_{t-j} - \hat{\mu})^T \end{aligned}$$

以下の連立方程式を $A(i)$ について解く:

$$C_j = \sum_{i=1}^{k} A(i) C_{j-i} \quad (j = 1, \cdots, k). \tag{6}$$

方程式 (6) の解を $\hat{A}(1), \cdots \hat{A}(k)$ とし，以下を計算

$$\begin{aligned} \hat{x}_t &:= \sum_{i=1}^{k} \hat{A}(i)(x_{t-k} - \hat{\mu}) + \hat{\mu} \\ \hat{\Sigma} &:= (1-r)\hat{\Sigma} + r(x_t - \hat{x}_t)(x_t - \hat{x}_t)^T \end{aligned}$$

図 7: SDAR アルゴリズム

本稿では，このアルゴリズムにおいて t 番目のデータまでを用いて得られる確率密度関数 (1) を，p_t と書く．

B　benefit の定義

変化点検出の精度を測るために，Fawcett and Provost による activity monitoring の枠組み [1] を参考に，benefit を導入した．具体的には変化点が生じた正しい時点 t^* に対し，時刻 t で閾値を超える警報が上げられたならば，その警報の benefit を以下の式で定義する．

$$\text{benefit}(t) \stackrel{\text{def}}{=} \begin{cases} = 1 - \frac{t-t^*}{20}, & \text{when } 0 \le t - t^* \le 20, \\ = 0, & \text{otherwize.} \end{cases}$$

これは遅延なしに検出すれば 1 になり，20 時点の遅延があれば 0 となる．すなわち，20 時点以内で検出した場合に正の benefit が与えられるようにしてある．また，ChangeFinder は原理的に，ある変化点に起因する警報をその変化点以前に上げることは無いので，変化点以前の警報は誤報とみなし，benefit を 0 としてある．

一方で GS と SC はバッチ的に検定を行うようなアルゴリズムである故，ある変化点より前に警報を与えることがある．そこで，GS と SC の評価には，次式の benefit を用いた．

$$\text{benefit}(t) \stackrel{\text{def}}{=} \begin{cases} 1 - \frac{|t-t^*|}{10}, & \text{when } |t - t^*| < 10, \\ 0 & \text{otherwize.} \end{cases}$$

二つの benefit の定義では，グラフを書いた場合正の部分の面積が等しいことに注意されたい．

図 3 に示した評価は，上記の benefit を実験用データに含まれる 99 個の変化点について計算して平均値を求めたものである．また，互いに近い時点の複数回の警報が出た場合，一つの警報のみに正の benefit を与えて評価した．詳しくは [4] を参照されたい．