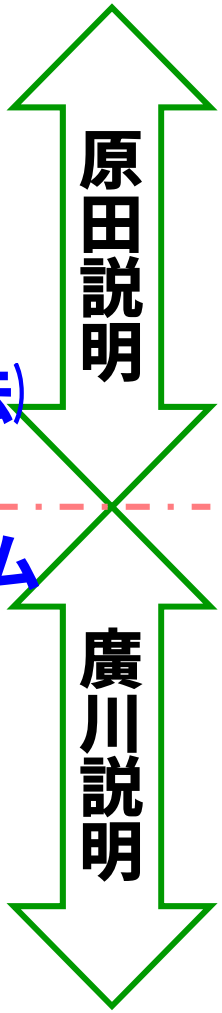
異常トラヒック検出アルゴリズムの比較検証

廣川　裕
NTT情報流通プラットフォーム研究所

これからの流れ

- アルゴリズムへの要求 (振り返り)
- 異常検知アルゴリズムいろいろ
  - 波形予測アルゴリズム
    - Holt-Winters**法** (HW:rrdtool)
    - Kalman-Hoffding **アルゴリズム** (KH:**原田法**)
  - 変化点検出アルゴリズム
    - Traffic Leap/sLump Detection **アルゴリズム** (TLD:**廣川法**)
- 各アルゴリズムの精度の比較
- まとめ



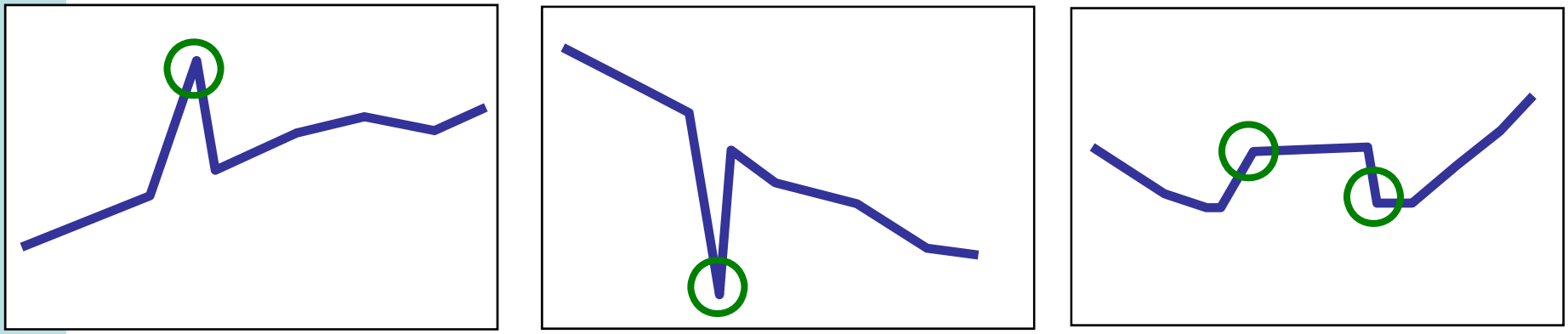
異常検知アルゴリズム紹介

変化点検出アルゴリズム

「急激な変化を検出」

変化点検出アルゴリズム

トラヒックの波形が急激に変化するポイントを検出する



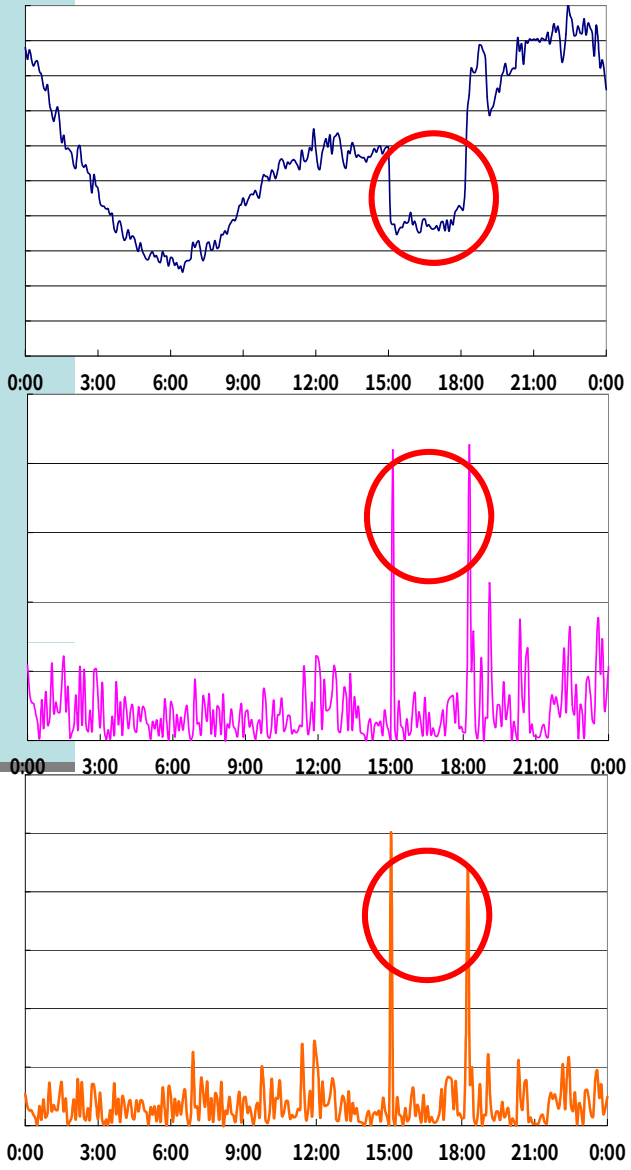
○：検出ポイント

変化点検出アルゴリズム

Traffic Leap/sLump Detectionアルゴリズム
(TLDアルゴリズム：廣川法)

TLDアルゴリズムは総務省委託研究
「次世代バックボーンに関する研究開発」による成果である。

TLDアルゴリズム (廣川法)



トラヒックのグラフから昼夜変動を取り除き、
閾値による異常の検出を容易にする。

トラヒックの流量ではなく
直前の値との差分をグラフ化

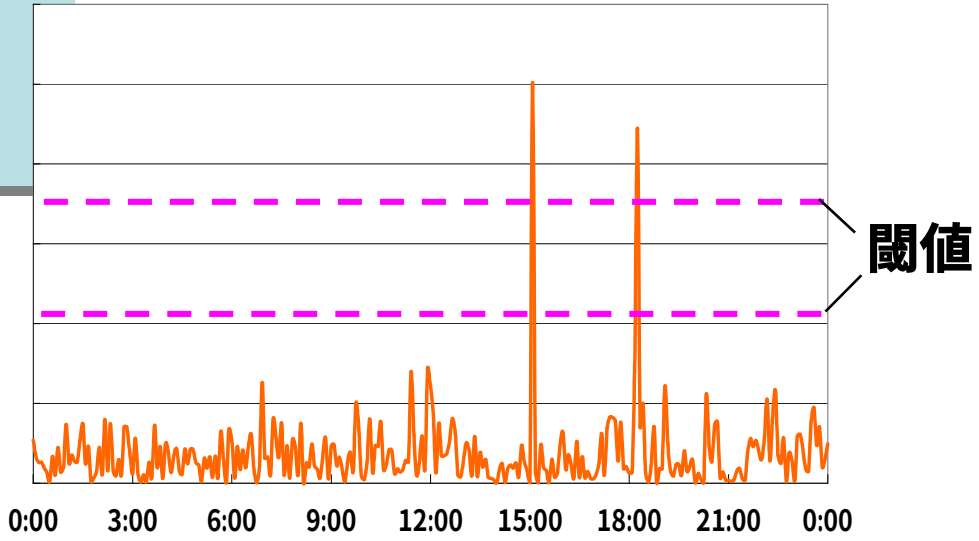
正規化を行い、昼夜変動の
影響を更に取り除く

急激な変動があったタイミングでのみ
大きなパルスが現れる

TLDアルゴリズムの動作条件

アルゴリズムを動作させるために必要な設定.
•許容量の感度パラメータ

周期性を考慮しないため、保持するデータ数も少なく短時間の学習期間で動作する。



メモ:

簡単な計算のみで動作するため、計算処理は軽い.
トラヒック波形毎の設定が無いため運用が楽。

各アルゴリズムの精度の比較

これまで紹介した異常検出アルゴリズムの精度を検証する

- Holt-Winters法(HW)
- Kalman-Hoffdingアルゴリズム(KH)
- Traffic Leap/sLump Detectionアルゴリズム(TLD)
- MFEEDツール(MF)

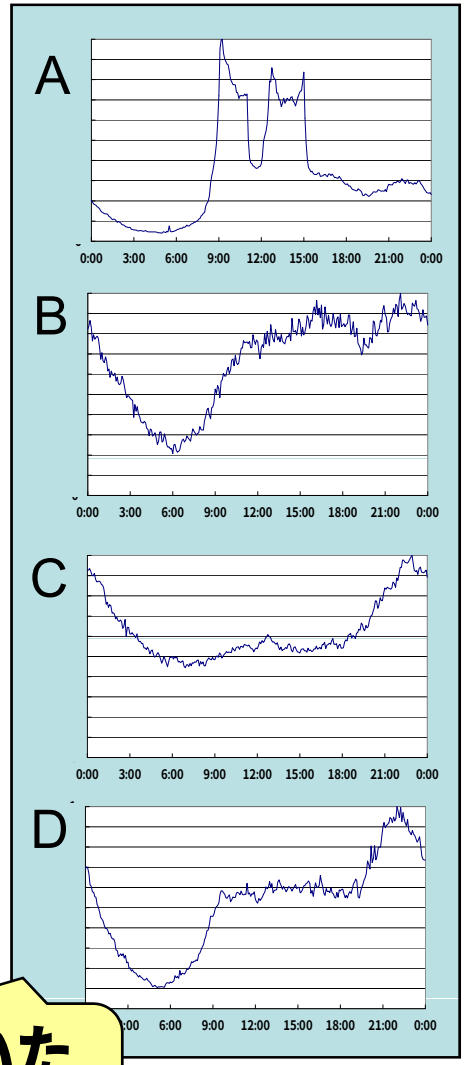
運用現場が望むツールの要件整理

	ポイント	指標
①	検知精度	見逃しや誤検知回数の抑制
②	祝日の対応	カレンダーを用意しなくも動く？
③	深夜早朝帯の対応	トラフィックが底の時間帯も誤検知しない？
④	解析に必要なデータ数(学習期間)	保持期間。より少ないデータで解析できる方が良い。
⑤	収束時の回復通知	事象発生時の通知だけでなく、回復時の連絡もできる？
⑥	パラメータ数	より簡単な設定で動くもの。

これらの要件に従って各アルゴリズムの比較検証を行う。

アルゴリズムの検証の方法

- JPNAPで観測された**4種類**、**2ヶ月分**の**5分間隔で集計**されたトラヒックデータを使用。
- オペレータの樽井さん、谷津さんを含めた発表者4人で協議し、**異常として検出したい箇所の正解リスト**を作成。
- 検出したい箇所は4種のデータ2ヶ月分合わせて**24箇所**。



検証に用いたトラヒックデータ

●見逃し

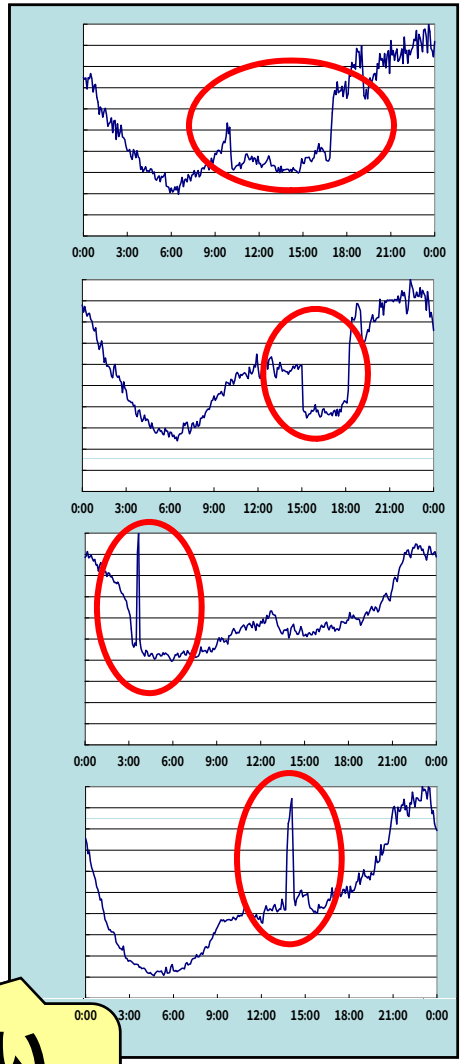
-正解リストの中で検出できなかった箇所

●誤検出

-正解リスト以外で検出された箇所

アルゴリズムの検証の方法

- JPNAPで観測された**4種類**、**2ヶ月分**の**5分間隔で集計**されたトラヒックデータを使用。
- オペレータの樽井さん、谷津さんを含めた発表者4人で協議し、**異常として検出したい箇所の正解リスト**を作成。
- 検出したい箇所は4種のデータ2ヶ月分合わせて**24箇所**。



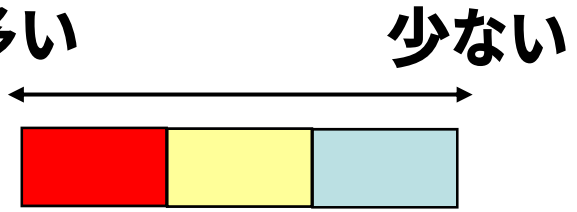
検出したい箇所の例

●見逃し
-正解リストの中で検出できなかった箇所

●誤検出
-正解リスト以外で検出された箇所

検知精度の比較(1/2)

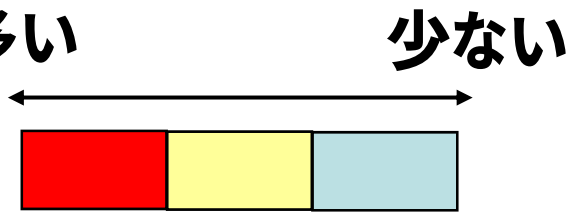
●見逃し発生回数



データ	データA	データB	データC	データD
Holt-Winters法	3回/4箇所	2/14	0/1	0/5
KHアルゴリズム	1/4	2/14	0/1	0/5
TLDアルゴリズム	0/4	3/14	0/1	0/5
MFEEDツール	2/4	1/14	0/1	0/5

見逃しの発生回数については大きな差は無い。

検知精度の比較(2/2)

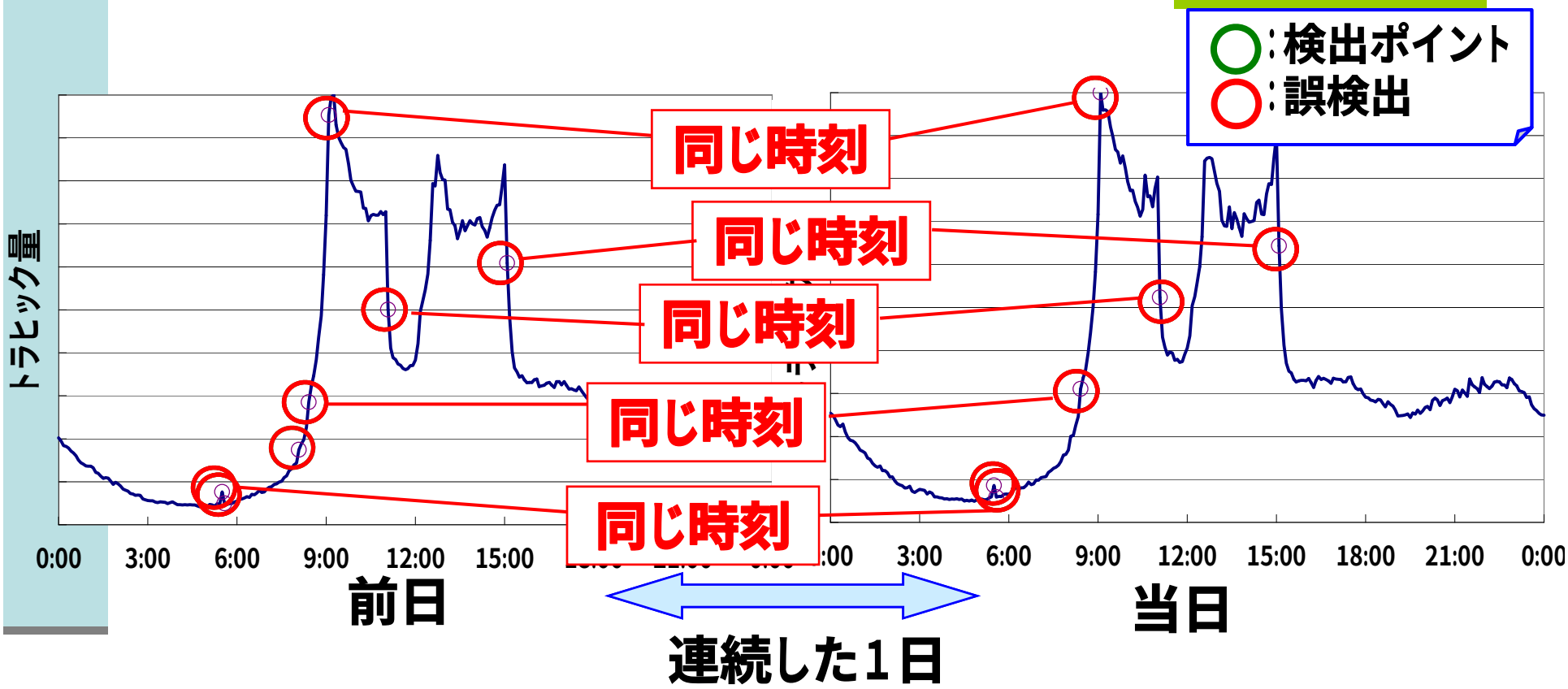


●誤検出の発生率

データ	データA	データB	データC	データD
Holt-Winters法	0.78回/日	0.59	0.51	0.86
KHアルゴリズム	0.63	0.05	0.02	0.32
TLDアルゴリズム	2.14	0.17	0.07	0.64
MFEEDツール	0.44	0.15	0.00	0.05

- HW法がやや誤検出が多い。
- TLDアルゴリズムはデータAで特に誤検出が多く発生。

周期変動を誤検出
(TLDアルゴリズム固有の問題)



●毎日同じ時刻に急変動が発生するトラヒックで発生

- TLDは周期性を考慮していない。

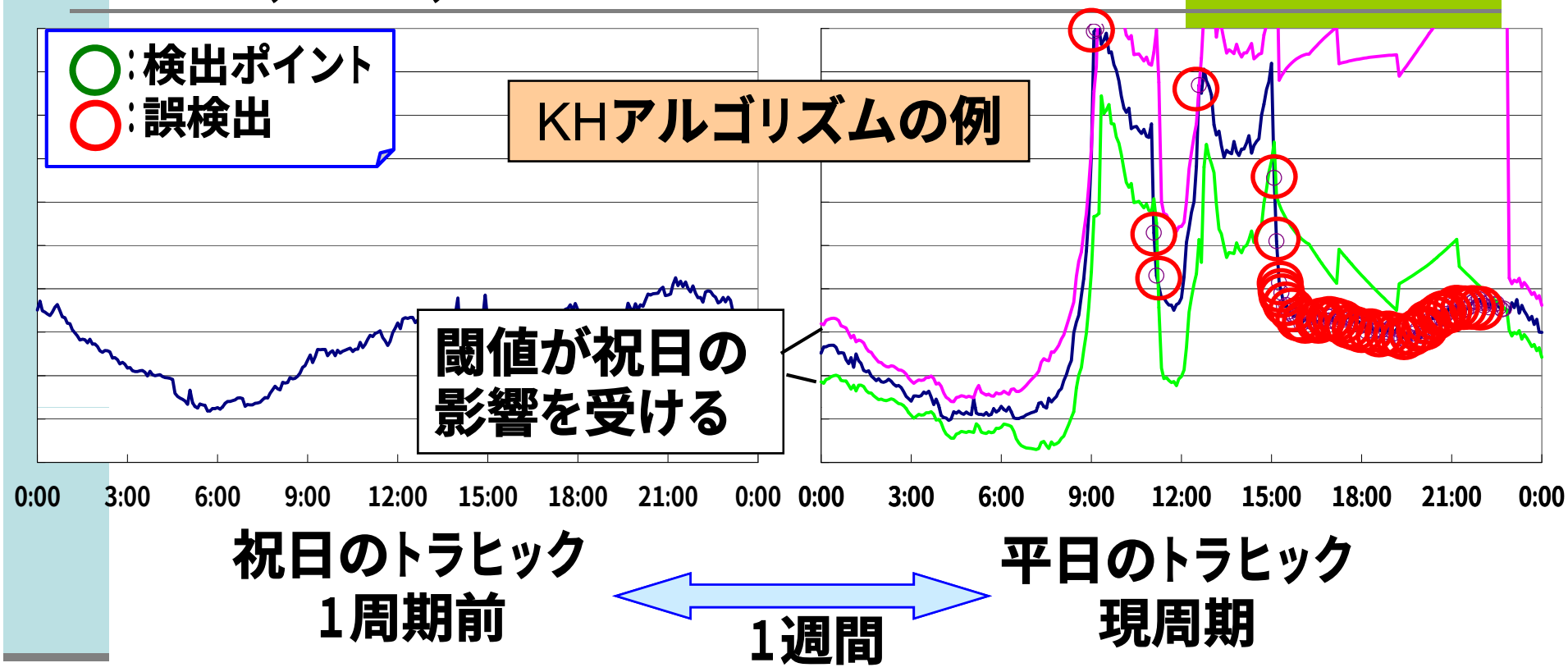
→これらの変動を全て検出してしまう。

各アルゴリズムの比較

	ポイント	HW	KH(原田)	TLD(廣川)	MF(梅井)
①	検知精度	△	○	△	○

周期変動を誤検出

祝日による誤検出
(HW, KH, MFアルゴリズム固有の問題)



●平日と休日・祝日で特徴が大きく異なるトラヒックで発生。

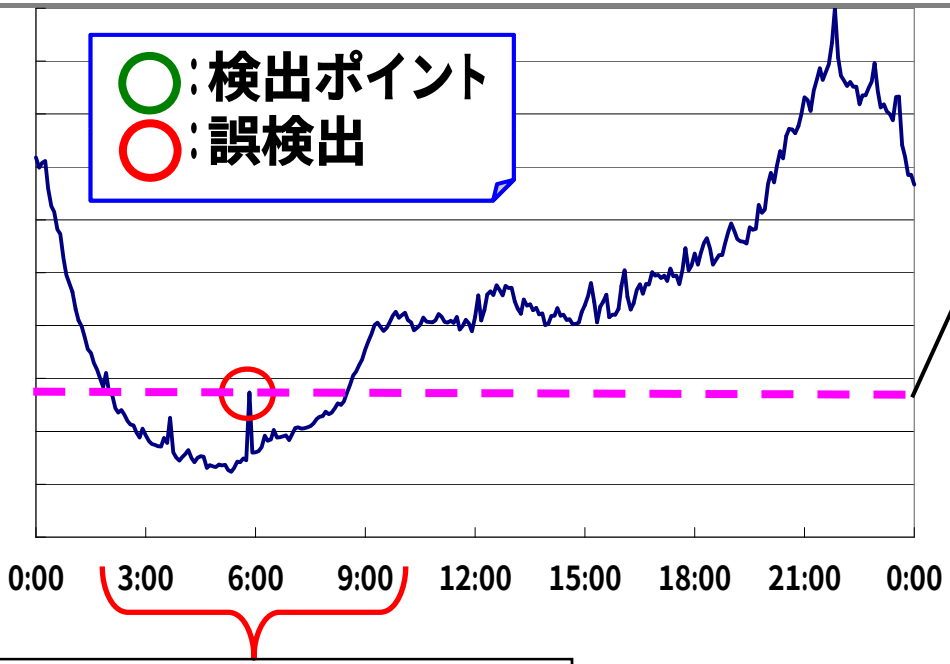
- 週変動を考慮したアルゴリズムでも、**祝日には対応できない。**

各アルゴリズムの比較

	ポイント	HW	KH(原田)	TLD(廣川)	MF(樽井)
①	検知精度	△	○	△	○
②	祝日の対応	×	×	○	×

祝日に対応していない

深夜早朝帯の誤検出 (全アルゴリズム共通の問題)



異常が検出されたが
流量は少ない

検出すべきか
意見が分かれる箇所

流量の減る時間帯

●深夜から明け方にかけての時間帯は誤検出とした。

-アルゴリズム的には検出すべき。
-このトラヒック増加を検出してもオペレータは嬉しくないのか？

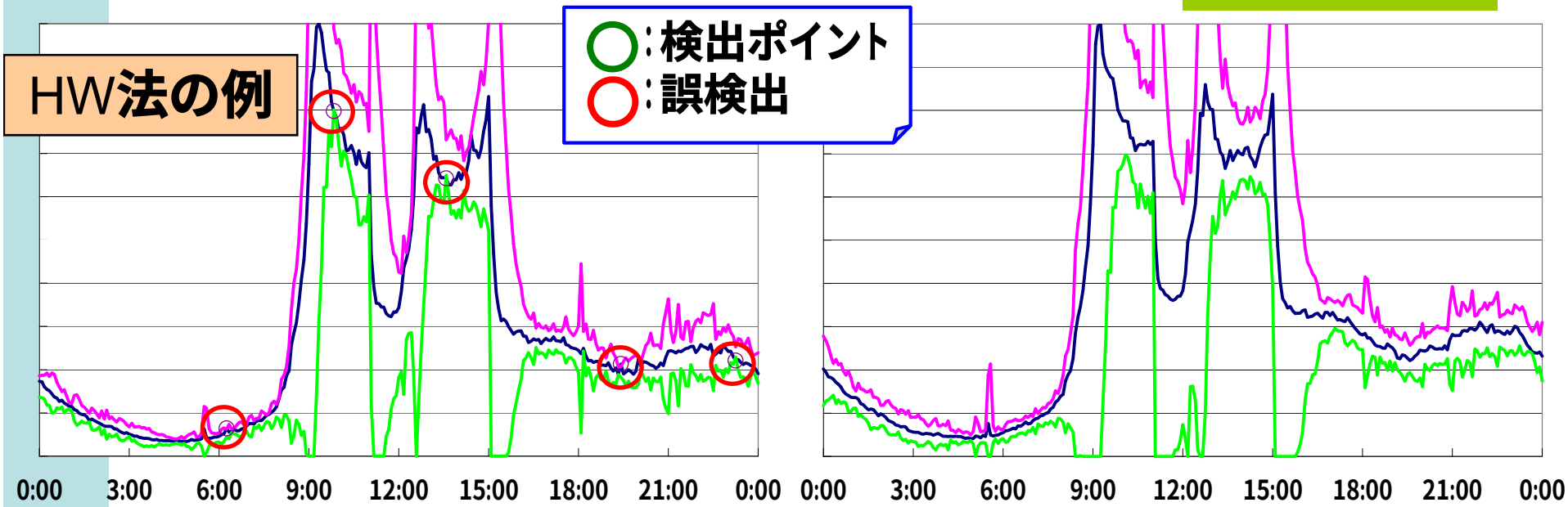
各アルゴリズムの比較

	ポイント	HW	KH(原田)	TLD(廣川)	MF(樽井)
①	検知精度	△	○	△	○
②	祝日の対応	×	×	○	×
③	深夜早朝帯の対応	×	×	×	×

どのアルゴリズムも
対応できていない

学習不足による誤検出
(HW, KHアルゴリズム固有の問題)

HW法の例



学習が不十分な場合

学習が十分な場合

●周期性を考慮したアルゴリズムは長い学習期間を要する。

-学習が不十分だと上図のように誤検出が発生する。

各アルゴリズムの比較

	ポイント	HW	KH(原田)	TLD(廣川)	MF(樽井)
①	検知精度	△	○	△	○
②	祝日の対応	×	×	○	×
③	深夜早朝帯の対応	×	×	×	×
④	解析に必要なデータ数(学習期間)	×	△	○	○

学習期間が必要
(KHの方が短い)

各アルゴリズムの比較

	ポイント	HW	KH(原田)	TLD(廣川)	MF(樽井)
①	検知精度	△	○	△	○
②	祝日の対応	×	×	○	×
③	深夜早朝帯の対応	×	×	×	×
④	解析に必要なデータ数(学習期間)	×	△	○	○
⑤	収束時の回復通知	×	△	△	×
⑥	パラメータ数	×	○	○	×

- **回復通知はKH、TLDで部分的に実現。**
- **KH、TLDはパラメータ設定の手間が少ない。**

まとめ

まとめ

- 現状では全ての要件を満たすアルゴリズムは無い
- アルゴリズムが検出する異常≠オペレータが思う異常
  - -深夜早朝帯のトラヒック増加は検出すべきか？
- その他、精度以外にも考えることもある
  - -大量のトラヒックデータを監視する時
    計算時間、必要とするメモリ量は大丈夫か？etc.

今後の取り組み

●KHアルゴリズム(原田法)

- トラヒック分割の箱長の自動調整法検討
- 祝日トラヒックへの対応

●TLDアルゴリズム(廣川法)

- 周期的に発生する急変動の対応
- 深夜早朝帯の誤検出への対応

おわり