

## JPIXトラフィックの短期予測

石橋, 弘隆 / ISHIBASHI, Hirotaka

---

(発行年 / Year)

2013-03-24

(学位授与年月日 / Date of Granted)

2013-03-24

(学位名 / Degree Name)

修士(工学)

(学位授与機関 / Degree Grantor)

法政大学 (Hosei University)

P377.5  
M34  
2012-12

2012年度 修士論文

## JPIX トラヒックの短期予測

法政大学大学院工学研究科  
情報電子工学専攻修士課程

学生証番号 11R4108

氏名 イシバシ ヒロタカ 石橋 弘隆

八名 和夫 教授





Year 2012

Master's Thesis

# A Method for JPIX Traffic Prediction

Hosei University Information and Electronics Engineering

Hiroataka Ishibashi

Advisor: Professor Kazuo Yana

## JPIX トラヒックの短期予測

現在, インターネットが社会インフラとして定着しその通信量は急速に増加し続けている. 総務省の調査によると平成 23 年度末時点で日本国内における一世帯あたりのインターネットの普及率は 86.0% となっており, 平成 24 年 5 月の時点で日本国内において発生するトラヒック総量は推定で 1.73Tbps に達している. トラヒック総量に関して言えば, 1 昨年同時期(平成 23 年 5 月時点)と比較して 14.1% 増となっている [1]. このように日本のネットワークトラヒックは増加の一途をたどっている. 急激なトラヒックの増加は, 通信事業者やインターネットサービスプロバイダ (ISP:Internet Service Provider) の安定的なネットワーク運用の妨げとなり, 大規模な通信障害の引き金にもなり得る. トラヒック増加による通信障害を防ぐには, 将来のトラヒックを予め予測する事が重要であり, 特に短期的には 1 日のピークのトラヒック量と, その時間を予測する事が重要である.

本稿では, ネットワークにおける通信障害を防ぐ足がかりとして, 任意のネットワーク上を流れるトラヒックを短期的に予測する研究を行った. 本稿での, 予測対象として日本のインターネットバックボーンの中核である, 日本インターネットエクスチェンジ [2] 上を流れるトラヒックを予測対象とし, 将来の 1 日分 (24 時間分) のトラヒックを予測した.

具体的な予測法としては, 一定期間収集したトラヒックデータに対して特異値分解を行うことにより直交成分を得て, その直交成分を元に短期的にトラヒックデータを予測する. また, 予測当日新たに得られる 5 分間隔のデータにより逐次的にトラヒック直交成分の重みを更新し最適な予測を得るアルゴリズムを提案する.

キーワード : ネットワークトラヒック, 日本インターネットエクスチェンジ, JPIX, 特異値分解

# Master's Thesis Overview Year 2012

## A Method for JPIX Traffic Prediction

Currently, Internet has established itself for social infrastructure and its traffic is increasing sharply. According to the investigation of Ministry of Internal Affairs and Communications, penetration rate of Internet is 86.0% per a household at year-end 2011 in Japan, and amount of Internet traffic occurred in Japan reaches 1.73Tbps(Tera bit per second) at May 2012. Regarding with the amount of Internet traffic, it is 14.1% higher compared with 1 year ago (May 2011)[1]. Thus the amount of Internet traffic is increasing in Japan. An exponential increase of Internet traffic disturbs telecommunication carriers' and ISPs'(Internet Service Provider) stable Network operation and even becomes a trigger of large-scale network disturbance. In order to prevent these incidences caused by increase of Internet traffic, it is important to make an accurate forecast and especially an estimation of peak volume and its time is important.

This paper proposes a method for making traffic predictions on any information network for the purpose of preventing network disturbances. In this paper, we studied JPIX(Japan Internet eXchange) traffic. JPIX is one of the most largest Internet backbone in Japan and our method is to make accurate daily prediction of the traffic volume. Singular value decomposition is applied to the daily JPIX traffic data for a certain time period (1-2 months) to extract the orthogonal time series underlying the traffic data. Then an adaptive algorithm is applied for updating the weight vector to improve the prediction accuracy.

Keywords: Network Traffic Prediction, JPIX, Japan Internet Exchange, Singular value decomposition

# 目次

<b>第1章</b>	<b>序論</b>	<b>1</b>
1.1	背景	1
<b>第2章</b>	<b>IX・JPIXについて</b>	<b>2</b>
2.1	IX	2
2.2	JPIXとは	4
2.3	トラヒック画像の数値化手法	5
2.3.1	数値化手順	5
<b>第3章</b>	<b>特異値分解</b>	<b>7</b>
3.1	特異値分解	7
3.2	複数時系列の特異値分解	8
3.3	JPIXの特異値分解	9
3.3.1	特異値分解を行うトラヒックデータ	9
3.3.2	特徴成分	10
3.3.3	元信号の復元	11
3.3.4	復元率	13
<b>第4章</b>	<b>JPIXのトラヒック予測</b>	<b>15</b>
4.1	予測トラヒック作成法	15
4.1.1	基本方針	15
4.1.2	予測手法	16
4.1.3	$U'$ の算出法	18
4.2	予測結果	20
4.2.1	予測トラヒックの作成	20
4.3	予測結果の統計的性質	22
4.3.1	予測の手順	22
<b>第5章</b>	<b>予測トラヒックの逐次更新</b>	<b>29</b>
5.1	逐次更新手法	29
5.1.1	基本方針	29
5.1.2	逐次更新手法	31
5.1.3	実際の2次元サーチ	33
5.1.4	四角形サーチにおけるアルゴリズム	35
5.1.5	探索範囲 $\alpha, \beta$ の算出法について	40
5.2	逐次更新の実例	44
5.2.1	逐次更新手法	44
5.2.2	実際のデータにおける逐次更新の実例	46
5.2.3	実際のデータにおける逐次更新の実例2	49
5.2.4	逐次更新法の有効性検証	51

第 6 章 忘却型逐次更新	54
6.1 基本方針	55
6.2 忘却度	57
6.3 忘却型逐次更新の実現	59
6.4 忘却型逐次更新の実例	61
6.5 忘却型逐次更新の有効性検証	66
6.6 逐次更新と忘却型逐次更新の性質	67
第 7 章 大規模通信障害	70
7.1 東日本大震災当日における JPIX のトラヒック	70
7.2 通信障害の検知	71
7.3 通信障害発生時における逐次更新	73
第 8 章 結論	77
謝辞	78
参考文献	79
付 録 A 特異値分解に用いたトラフィックデータ	80
付 録 B 特徴成分	85
付 録 C 各成分に対する $U$	90
付 録 D 相対誤差と時刻の関係	95
付 録 E 相対誤差と時刻の関係	99

# 第1章 序論

我が国では、平成23年末の時点で一般世帯のインターネットが普及率が86.0%[1]、従業員100人以上の企業におけるインターネット普及率が98.8%[1]となっており、インターネットは公私ともに日本人々の生活を支えていると言える。そのような状況の中で、インターネットを利用したシステムやアプリケーションが多数開発されている。インターネットを利用したシステムには、企業の業務システムや証券取引システムといった高い信頼性が求められるものもあり、通信事業者にとって大規模な通信障害を起こす事は許されない状況になりつつある。今後、スマートフォンの普及による更なるトラフィック増加が予想される中で、安定したネットワークの運用を行う為に、精度の高いトラフィックの予測手法はますます重要になってくると言える。

## 1.1 背景

トラフィックの増加が一因となって、大規模な通信障害を引き起こしたと言われている例がある。それは、2007年5月15日～16日にかけて、NTT東日本のフレッツ・サービスに大規模障害が発生したものである。障害が起きたのは、23区を除く東京都、北海道、青森県、岩手県、宮城県、秋田県、山形県、福島県、茨城県、栃木県、群馬県、新潟県、山梨県、長野県の14都道県であり、合計約239万計約のユーザがインターネットに接続できないという状況になった。公式にはソフトウェアのバグが原因という事になっているが、契約ユーザのトラフィックが増加していた事も障害の原因[2]になったと言われている。

そのような、現実に発生し得る通信障害を未然に防ぐ為の足がかりとして、本研究では日本インターネットエクスチェンジ(以降JPIX)上を流れるトラフィックを予測する手法を提案する。解析対象としてJPIXを選択したのは2つの理由がある。

1つ目の理由は、JPIXは多数のISP間を相互接続している日本で最大規模の商用IXであるという事である。その規模の大きさから、非常に多くのトラフィックが日本各地から集められJPIX上を流れるようになっている。その為、JPIX上を流れるトラフィックは日本のインターネットトラフィックの縮図であると言える。本研究では、日本のインターネットトラフィックの特徴を示す大規模トラフィックデータを扱いたかった為解析対象とする1つの理由となった。

2つ目の理由は、比較的容易にトラフィックデータが手に入るという事である。JPIXは自社のホームページ上でリアルタイムのトラフィックグラフを公開しており、誰でもグラフを入手する事ができる。一般的に大規模ネットワークのトラフィックデータを手に入れる事は困難であるが、JPIXでは貴重なトラフィックデータを誰もが参照できるように公開しており、比較的容易に収集する事が可能である。それがJPIXを解析対象とした2つ目の理由である。

## 第2章 IX・JPIXについて

本章では, IX についての説明, JPIX についての説明, JPIX データの入手法について解説する.

### 2.1 IX

IX とはインターネットエクスチェンジ (Internet Exchange) の略称である. IX は効率良く複数の ISP を接続する為の中継点としての役割を担っている. IX の役割を分かり易く説明すると以下のような説明になる. インターネットとは, 各 ISP が管理するネットワークを相互に接続したものだと思える事が出来る. インターネット上では, 全ての端末同士が何らかの形で繋がっていないと成らない為, 各 ISP が管理するネットワーク同士の接続性も保障されなければならない. そこで, 仮に 8 つの ISP 同士を相互に接続すると図 2.1 のようになる.

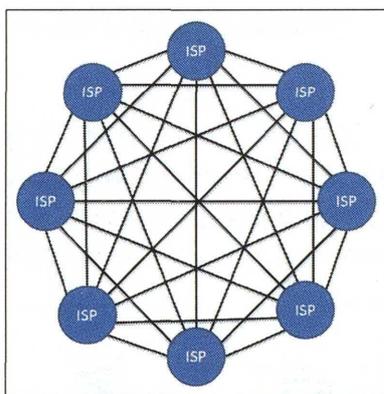


図 2.1: IX 無しで ISP を相互に接続する場合

図 2.1 を見れば分かるように, たった 8 つの ISP 同士を相互接続するのに多数の接続線が必要になり, この例では 28 本の接続線が必要となっている. ISP の数を  $n$  とした時, 相互接続に必要な接続線は  $n(n-1)/2$  で求められる. 日本には大小併せて数千社の ISP が存在すると言われているので, 全ての ISP の接続性を保障するのは非常に難しい. そこで, IX を用いて 8 つの ISP を接続した場合を図 2.2 に示す.

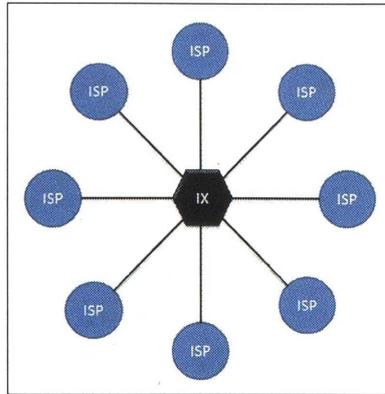


図 2.2: IX 無しで ISP を相互に接続する場合

図 2.2 は, 図 2.1 に比べて接続線が少なくなっており, 構成もシンプルになっている事が分かる. このことより, IX を用いた方が効率よく ISP を相互接続できると分かる. 一般的に, IX を利用する事で以下のようなメリットが得られる.

- 他の ISP との接続性を高いレベルで保障できる.
- IX と接続されている ISP 同士であれば, データの受け渡しの際に経由するネットワークが減り, 伝送遅延と伝送損失を減らす事が出来る.
- 他の ISP と接続する為の接続線が減る為, 接続線を引く為のコストが減り, 管理する為のコストも少なくて済む.

ISP 間でデータの受け渡しを行う事をピアリングと呼ぶ. IX を利用したピアリングをパブリッシュ・ピアリングと呼び, IX を利用せずにピアリングを行う事をプライベート・ピアリングと呼ぶ. 現在のインターネットは非常に複雑な構成になっており, 図 2.3 に示すようにパブリッシュ・ピアリングとプライベート・ピアリングが混在する状態になっている.

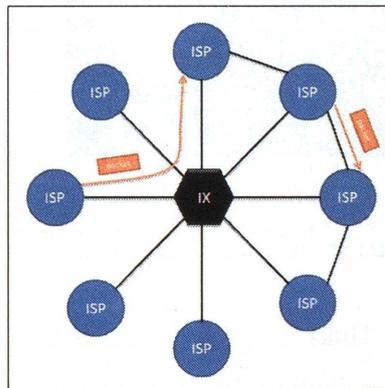


図 2.3: パブリッシュ・ピアリングとプライベート・ピアリングの混在

## 2.2 JPIX とは

JPIX は 1997 年 7 月 10 日に KDDI, インターネット総合研究所, シスコシステムズ, 富士通など 16 社の協力で設立された日本最初の商用インターネットエクスチェンジ会社である。現在では, 取扱いトラフィック及び接続顧客数ともに日本最大の IX であり, 日本のインターネットの心臓部として重要な役割を担っている。JPIX では 5 分間隔のトラフィックデータを画像データとしてのみ公開している。JPIX のトラフィックは, 日本のインターネットトラフィックの特徴を反映するようになっており, 日本の人々のインターネット利用の特徴を表すようになってきている。平日と休日ではインターネットの利用率が変わってくる為, JPIX のトラフィックも平日と休日ではその特徴が異なっている。図 2.4 に典型的な平日のトラフィックを, 図 2.5 に典型的な休日のトラフィックをそれぞれ示す。

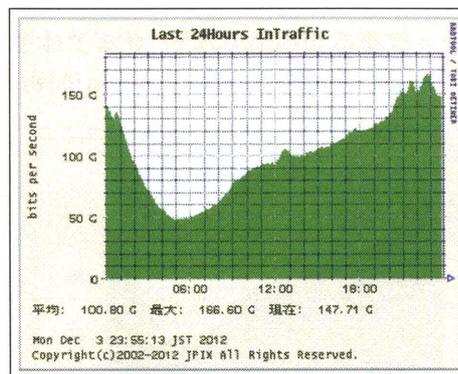


図 2.4: 平日における JPIX のトラフィック (2012 年 12 月 3 日 (月))

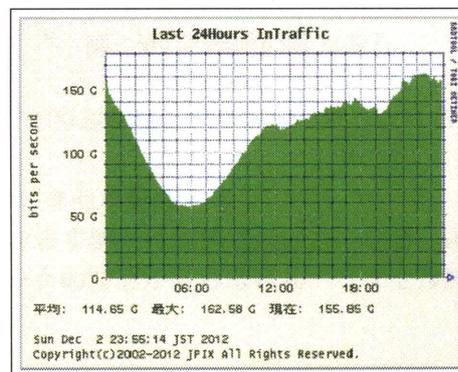


図 2.5: 休日における JPIX のトラフィック (2012 年 12 月 2 日 (日))

図 2.4 と図 2.5 を比較すると, 11 時から 17 時にかけて休日の方がトラフィックが高くなる傾向である事が分かる。これは日本の人々のライフスタイルとして, 平日は学校や会社にいる為お昼の時間帯にインターネットが利用できないのに対して, 休日はお昼の時間帯にもインターネットを利用できる事が大きく影響していると考えられる。以上が基本的な JPIX のトラフィックの特徴である。

## 2.3 トラフィック画像の数値化手法

前述の通り,JPIX は自社のホームページ上で画像データしか公開していない. また, その画像データは図 2.4, 図 2.5 の形式で公開されている. JPIX のトラフィックについて解析する為には, トラフィックの数値データが必要になる為画像処理手法を用いて, 画像データから数値データへと変換を行った. セクション 2.3 では, 数値データへの変換法を簡単に説明する.

### 2.3.1 数値化手順

数値化は以下の 3 ステップによって実現される.

画像から平均値を読み取る.

画像データには平均値が示されており, それを読み取る事でトラフィック部分のスケールが読み取れる. 図 2.6 に平均値の例を示す.

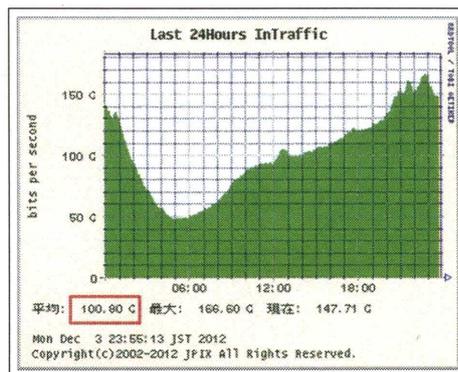


図 2.6: 平均値の読み取り

図 2.6 の例では, 平均値が 100.80Gbps である事が読み取れる.

画像のトラフィック部分のピクセルを数える.

画像データで, トラフィックを表す部分は緑色で示されている. 緑色の部分のピクセル数を読み取る事で, 1 日のトラフィックの波形が読み取れる. ピクセル数から波形を読み取ると, 図 2.7 のようになる.

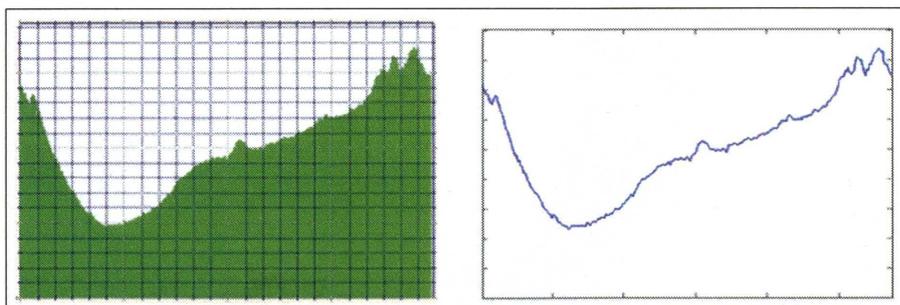


図 2.7: 波形の読み取り

波形の平均値を、画像から読み取った平均値に合わせる。  
ピクセル数で読み取った波形は、ボリュームが実際のトラフィックと異なっているので、初めに読み取ったトラフィックの平均値を使い、実際のトラフィックのボリュームと合わせる。ボリュームを合わせると、図 2.8 のようになる。

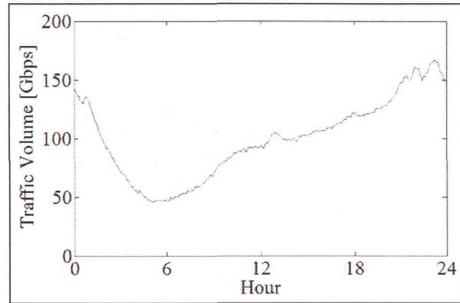


図 2.8: ボリューム合わせ

## 第3章 特異値分解

JPIX のトラヒックを予測する上で、特異値分解を用いた。この章では、特異値分解に関する説明と、実際に JIPX のトラヒックを特異値分解した結果を示す。

### 3.1 特異値分解

特異値分解とは、観測される複数の時系列信号を無相関で直交な信号に分解する事である。複数の時系列信号に対して特異値分解を実行する事で、複数の時系列信号に共通して見られる特徴を抽出する事が出来る。多くの場合、特異値分解は主成分分析に用いられ、本研究においても毎日観測されるトラヒックから主要な成分を抽出する為に用いている。トラヒックに対して特異値分解を行う事で、どの特徴が主要な特徴であるかを判定し、予測に用いるべき特徴を決定するのに用いた。特異値分解を式で説明すると、以下のようなものである。任意の  $m \times n$  行列  $X$  は次のように分解する事が出来る。

$$X = U\Sigma V^T \quad (3.1)$$

$U$  : 行列  $X$  の入力基底となる  $m \times m$  正規直交行列

$V$  : 行列  $X$  の出力基底となる  $n \times n$  正規直交行列

$\Sigma$  :  $\mu_1, \mu_2, \dots, \mu_K$ ,  $K = \min(m, n)$  を主対角要素にもつ  $m \times n$  対角行列

$\mu$  :  $\mu_1, \dots, \mu_k, \dots, \mu_K$  となる行列  $X$  の特異値

ここで、 $K \geq k$  の場合、 $U, V$  はそれぞれ

$$UU^T = I \quad (3.2)$$

$$VV^T = I \quad (3.3)$$

を満たす  $k \times m, k \times n$  の行列となる。ここで、 $I$  は単位行列である。

### 3.2 複数時系列の特異値分解

複数時系列信号  $X^{(m)}[n]$  ( $m = 1, \dots, M$ ) ( $n = 1, \dots, N$ ) が観測されているとする。ここで、 $m$  は観測されている時系列信号の本数を表しており、 $n$  は一つの信号あたりの点数を表している。  $X^{(m)}[n]$  が  $m \times n$  行列であるとして特異値分解を行うと、式 (3.1) より

$$X^{(m)}[n] = U\Sigma V^T \quad (3.4)$$

が成り立つ。ここで、特異値の階級を表す  $r$  ( $r = 1, \dots, R$ ) を定義し式 (3.4) の右辺の要素を用いると、

$$X^{(m)}[n] = \begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ u_{21} & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ u_{m1} & u_{m2} & \cdots & u_{mn} \end{pmatrix} \begin{pmatrix} \mu_1 & 0 & \cdots & 0 \\ 0 & \mu_2 & \cdots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ 0 & \cdots & 0 & \mu_r \end{pmatrix} \begin{pmatrix} v_{11}^T & v_{12}^T & \cdots & v_{1n}^T \\ v_{21}^T & v_{22}^T & \cdots & v_{2n}^T \\ \vdots & \vdots & \ddots & \vdots \\ v_{r1}^T & v_{r2}^T & \cdots & v_{rn}^T \end{pmatrix} \quad (3.5)$$

のように表す事ができる。

### 3.3 JPIXの特異値分解

本稿で提案する短期予測法は特異値分解を基礎としている.  $k$  日目 ( $k = 1, \dots, K$ ) 1 日分のトラフィックデータ (5 分間隔)

$$X_k[n], \quad n = 1, \dots, N \quad (3.6)$$

を  $k$  行目とするデータ行列を  $X$  とする.  $X$  の特異値分解によりトラフィック時系列の基礎となる正規直交信号を要素とする行列  $V$  を得る. すなわち特異値分解により  $X$  は

$$X = U\Sigma V^T \quad (3.7)$$

と分解される.  $S = \Sigma V^T$  と置き, 行列  $S$  の  $k$  行目を

$$s_k[n] \quad n = 1, \dots, N \quad (3.8)$$

とすれば, 各トラフィックデータは

$$X_k[n] = \sum_{l=1}^k u_l s_l \quad (3.9)$$

と表される.  $S$  には, 入力した複数時系列信号から抽出した特徴成分が格納されており, 入力した時系列信号の数と同じ数だけ特徴成分を得る事ができる. 今回は, 特異値分解の例として 2012 年 11 月 1 日~28 日までの 28 日分のトラフィックを入力信号  $X$  として, 特徴成分  $S$  を得た.

#### 3.3.1 特異値分解を行うトラフィックデータ

本研究では, 2012 年 11 月 1 日~28 日までの 28 日分のトラフィックを特異値分解した. ここでは代表例として, 2012 年 11 月 1 日~4 日までのトラフィックデータを次の図 3.1~図 3.4 に示す. 残りの 2012 年 11 月 5 日~2012 年 11 月 28 日のトラフィックデータは付録 A に示す.

トラフィックデータは, 特異値分解の式  $X = U\Sigma V^T$  のうち  $X$  にあたる. 1 日分のトラフィックが 1 本分の入力信号であり, 1 日当たり 287 点分のデータである. その為  $X$  は,  $287 \times 28$  行列となる.

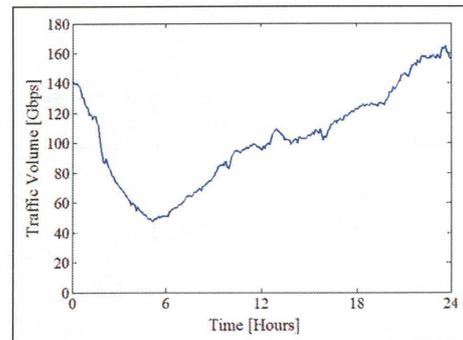
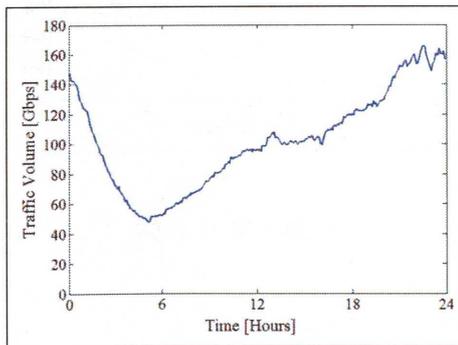


図 3.1: 2012 年 11 月 1 日 (木) のトラフィック 図 3.2: 2012 年 11 月 2 日 (金) のトラフィック

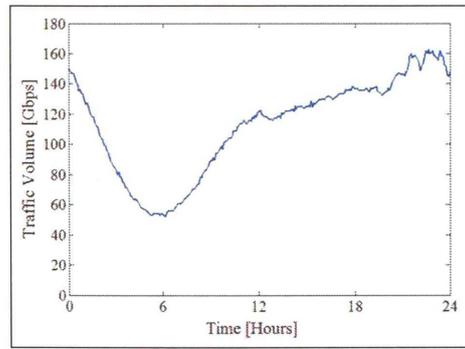
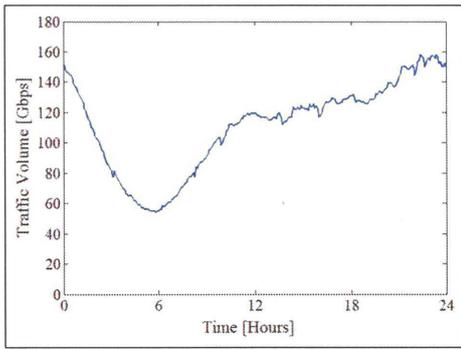


図 3.3: 2012年11月3日(土)のトラヒック 図 3.4: 2012年11月4日(日)のトラヒック

### 3.3.2 特徴成分

特徴成分  $S$  のうち最も重要度が高いものから第1成分, 第2成分,... と呼ぶ事とする. また, 今回は28本の時系列信号を入力しているので, 特徴成分  $S$  は28本得られている. ここでは代表例として, 第1成分~第4成分までの特徴成分を図3.5~図3.8に示す. 第5成分~第28成分までは付録Bに示す.

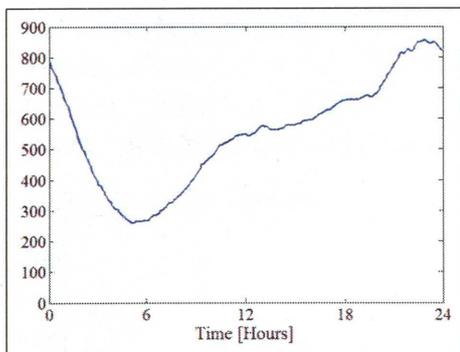


図 3.5: 第1成分

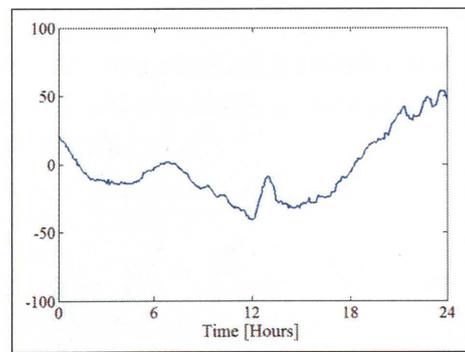


図 3.6: 第2成分

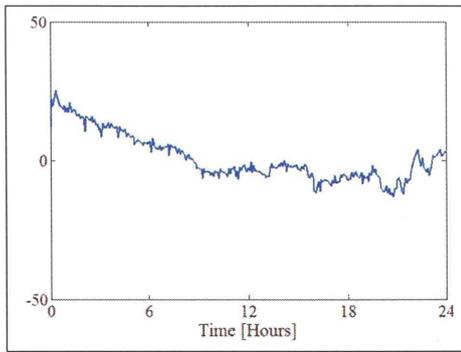


図 3.7: 第 3 成分

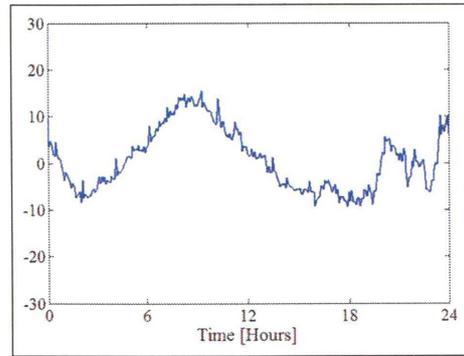


図 3.8: 第 4 成分

図 3.5 を見ると、特異値分解前のトラヒックの特徴に近い事が分かる。第 1 成分は、入力した信号の特徴のうち最も重要な特徴を表す為、この例のように元の信号に近い特徴を示す。また、後半の特徴になるほどノイズのようになり、元の信号の主要な特徴を表さなくなっている事が分かる。仮にこれら 28 個の特徴を用いて、将来のトラヒックを予測するとした場合、初めの数個の特徴成分だけを用いるべきであり、後半の特徴成分は主要な特徴では無い為用いるべきではない。

### 3.3.3 元信号の復元

特異値分解をして得られた  $US$  (または  $U\Sigma V^T$ ) を用いる事で、元の信号  $X$  を復元する事が可能である。今回の特異値分解の例では、28 日分のデータを特異値分解した。28 日分のトラヒックデータの基礎となる特徴  $S$  は全て共通であり、図 3.5～図 3.8 に示した通りである。各日でそれぞれトラヒックが異なるが、その違いをもたらしているのは  $U$  である。  $U$  は、各日における各特徴成分の大きさを示すものであり、各特徴成分の大きさの違いが各日のトラヒックの違いをもたらしている。説明が難解であるので、前述の特異値分解の例における実際の  $U$  を示して説明する。代表例として、各 1 成分～第 4 成分に対する  $U$  を次の図 3.9～図 3.12 に示す。また、第 5 成分～第 28 成分に対する  $U$  は付録 C に示した。

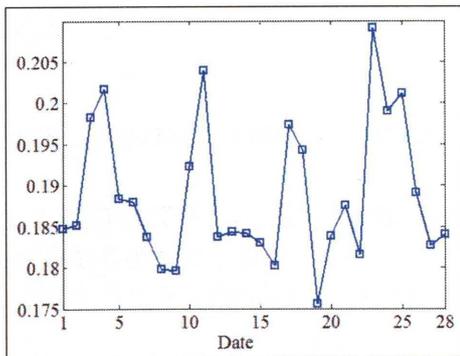


図 3.9: 第 1 成分に対する  $U$

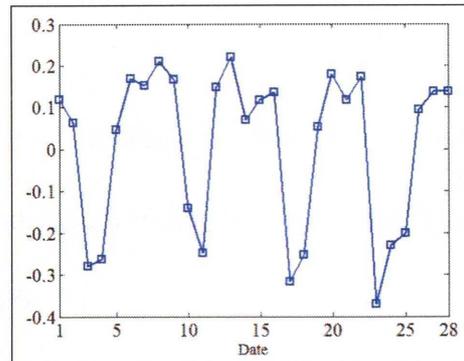


図 3.10: 第 2 成分に対する  $U$

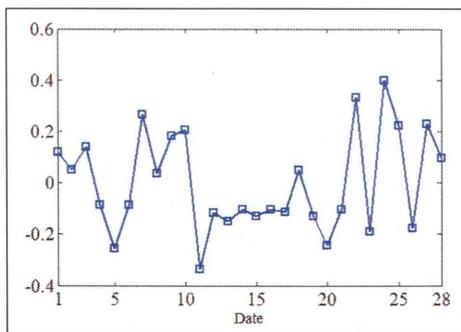


図 3.11: 第 3 成分に対する  $U$

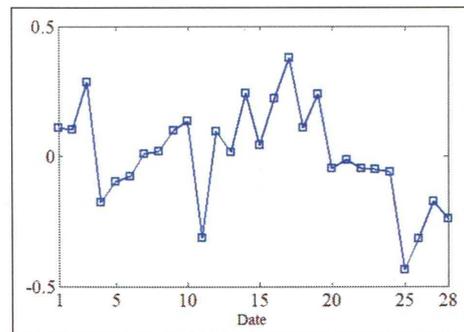


図 3.12: 第 4 成分に対する  $U$

図 3.9 を見ると、 $U$  の値が高い点が周期的に存在する事が読み取れる。図 3.9 の横軸は特異値分解されたトラヒックの日付と結びつけられており、1 日目の 2012 年 11 月 1 日は木曜日である。その為  $U$  の横軸は 1 点目から順番に、木、金、土、日、月、火、水、...、水を表している。土曜日と日曜日のトラヒックは、3,4,10,11,17,18,24,25 日目であり、図 3.9 の  $U$  が高い点と一致するようになっている事が分かる。これは、土曜日・日曜日は平日に比べて第 1 成分が大きい事を示している。

図 3.10 を見ると、先ほどとは逆に土曜日・日曜日には  $U$  の値が低くなっている事が読み取れる。これは、土曜日・日曜日は平日に比べて第 2 成分が小さい事を示している。

第 3 成分以降になると、平日と休日における顕著な違いは見取れない事が分かる。

このように  $U$  は、各日における各特徴成分の大きさを表しており、 $U$  に示された比率通りに各特徴成分  $S$  を足し合わせる事で元信号の復元が可能である。

例えば、元トラヒックの 1 日目である 2012 年 11 月 1 日 (木) を復元する場合は式 (3.10) のようになる。

$$\text{第 1 成分} \times 0.185 + \text{第 2 成分} \times 0.120 + \text{第 3 成分} \times 0.119 + \dots + \text{第 28 成分} \times 0.103 \quad (3.10)$$

式 (3.10) で出てきた 0.185, 0.120, 0.119, ... , 0.103 という数字は、各  $U$  の画像の 1 点目から読み取る事ができる。

また、元トラヒックの 4 日目である 2012 年 11 月 4 日 (日) を復元する場合は式 (3.11) のようになる。

$$\text{第 1 成分} \times 0.202 + \text{第 2 成分} \times (-0.263) + \text{第 3 成分} \times (-0.085) + \dots + \text{第 28 成分} \times (-0.061) \quad (3.11)$$

式 (3.11) で出てきた 0.202, -0.263, -0.085, ... , -0.061 という数字は、各  $U$  の画像の 4 点目から読み取る事ができる。

以上が元トラヒックの復元法についての説明である。

### 3.3.4 復元率

3.3.3章で説明したように、全ての特微成分  $S$  と、各特微成分の大きさ  $U$  を用いる事で元信号の完全な復元が可能である。それに対して、復元の際に全ての特微を用いなかった場合、元信号の完全な復元は出来ないが、元信号の特微をある程度まで表現する事は可能である。特異値分解して得られた特微成分が  $n$  個であった時、第1成分から第  $m$  成分 ( $n \geq m$ ) までを用いて元信号を復元した時に、復元信号が元信号をどれだけ表現出来ているかを復元率と呼ぶ。

復元率は、特異値分解の式  $X = U\Sigma V^T$  のうち  $\Sigma$  の内容から算出する事ができる。特異値分解をして、 $m$  個の特微成分を得た時、1 個目の特微成分から  $n$  個目 ( $m \geq n$ ) までの特微成分を用いて元信号を復元した時の復元率は、式 (3.12) のように表される。

$$\frac{\mu_1 + \mu_2 + \dots + \mu_m}{\mu_1 + \mu_2 + \dots + \mu_n} \times 100 \quad [\%] \quad (3.12)$$

式 (3.12) において  $m = n$  であった場合、式 (3.13) のようになる。

$$\frac{\mu_1 + \mu_2 + \dots + \mu_m}{\mu_1 + \mu_2 + \dots + \mu_m} \times 100 = 1 \times 100 = 100 \quad [\%] \quad (3.13)$$

式 (3.13) は、全ての特微成分を用いる事で、元信号を完全に復元できることを表している。

前述の特異値分解の例において得られた  $\Sigma$  を用いて、復元率を求めた場合のグラフを図 3.85 に示す。

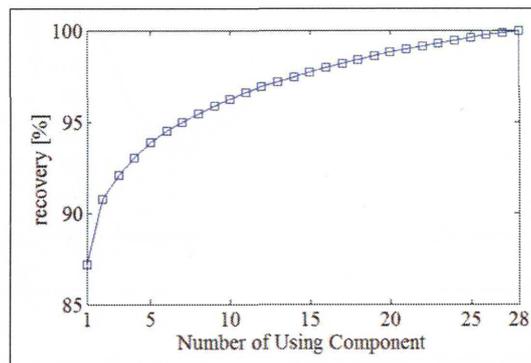


図 3.13: JPIX トラフィックにおける復元率

図 3.85 を見ると、第1成分のみで約 87% の復元率があり、第1成分と第2成分を用いると約 91% の復元率がある事が分かる。この事から、JPIX を表現するのに重要な特微成分は初めの数個の特微成分である事が分かる。ここで実際に、初めの数個の特微成分を用いて元信号を復元した例をしめす。

前述の特異値分解の例において、2012年11月1日(木)のトラフィックを復元する例を示す。第1成分のみを用いて復元する場合、式 (3.14) のようになる。

$$\text{第1成分} \times 0.185 \quad (3.14)$$

第1成分と第2成分を用いて復元する場合、式 (3.15) のようになる。

$$\text{第1成分} \times 0.185 + \text{第2成分} \times 0.120 \quad (3.15)$$

また、第1成分のみで復元した場合のトラフィックを図3.86に、第1成分と第2成分を用いて復元した場合のトラフィックを図3.87に示す。

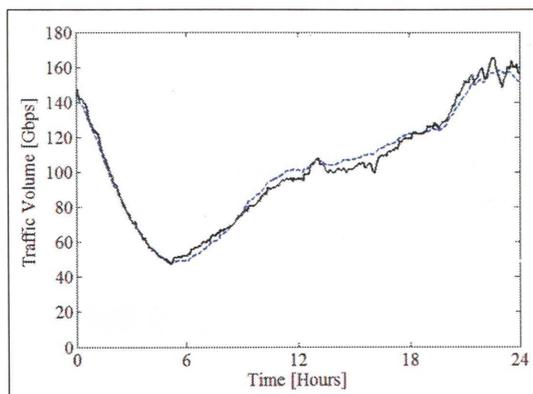


図 3.14: 第1成分のみの復元

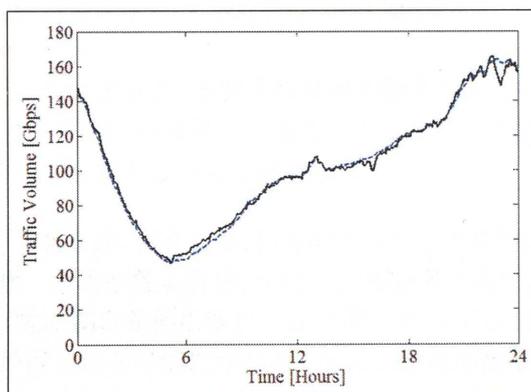


図 3.15: 第1成分と第2成分を用いた時の復元

図3.86を見ると、第1成分のみで元トラフィックの特徴の大部分を表現できている事がわかる。図3.85の復元率を見ても、第1成分のみで元トラフィックの約87%を表現できると読み取れる。

図3.87を見ると、図3.86より精密に元トラフィックの特徴を表現できている事がわかる。これは、第2成分が加わる事により、より細かい部分の特徴まで表現できるようになった為である。図3.85の復元率を見ると、第1成分と第2成分を用いる事で元トラフィックの約91%を表現できると読み取れる。

本研究での予測は、上に示したような復元の操作を基礎としている。近い将来におけるトラフィックの特徴成分  $S'$  が、現在のトラフィックの特徴成分  $S$  に近いと仮定すると、将来における各特徴成分の大きさ  $U'$  を推定する事で、トラフィックの予測を行う事が可能である。近い将来における  $U'$  は、現在のトラフィックを特異値分解して得られた  $U$  から推定を行う。

以上が復元率についての説明である。

## 第4章 JPIXのトラヒック予測

この章では、予測トラヒックの作成手法を説明し、実際に予測トラヒックの作成を行う。また、予測トラヒックと実際に観測されたトラヒックを比較する事で、予測の評価も行う。

### 4.1 予測トラヒック作成法

#### 4.1.1 基本方針

過去28日分のJPIXトラヒックデータを用いて、将来7日分の予測トラヒックデータを作成する。28日分のトラヒックを用いる理由としては、以下の3つの理由が挙げられる。

- $7 \times 4 = 28$  より、28日分のデータであれば、同じ曜日のデータが4日分ずつ含まれているから。
- 用いるデータ数が少なすぎると、本質的な特徴を捉えられない可能性があるから。
- JPIXトラヒックは長期的には変動しており、用いるデータ数が多すぎると、現在のトラヒックとは上手く合わなくなってしまうから。

1つ目の理由についてであるが、例えば30日分のトラヒックを用いる場合を考えると、月曜・火曜・水曜・木曜・金曜が各4日分、土曜・日曜が各5日分含まれるデータを特異値分解した場合、土曜・日曜の割合が他の曜日に比べ高くなってしまい、休日の特徴を通常より多く示す特徴成分が得られる可能性がある。そのような事態を防ぐ為に、各曜日が含まれる日数を揃えた。

2つ目の理由についてであるが、特異値分解によって特徴成分を得ようとした場合、ある程度のデータ数を用意しなければ上手く特徴を抽出できない場合が考えられる。本研究では、約1カ月分に相当する28日分のトラヒックを用意すれば十分であると判断した。

3つ目の理由についてであるが、以下の図4.1に示すようにJPIXのトラヒックは長期的に見て大きく変動している。その為、あまりに長期のトラヒックを用いてしまうと変動部分を取り今でしまい、上手く予測データが作成できなくなってしまう。本研究では、28日分のトラヒックであれば変動が無視できるレベルであると想定し、28日分のトラヒックを用いるのが妥当であると判断した。

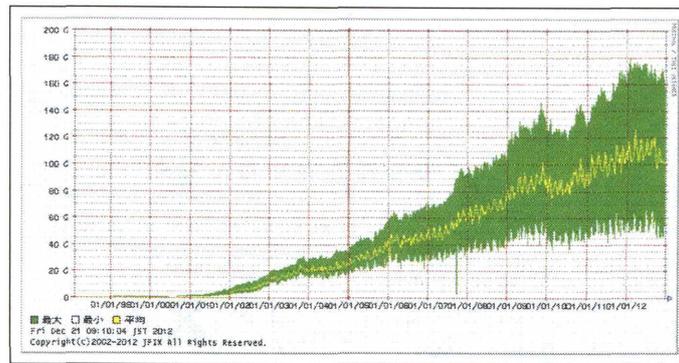


図 4.1: 1999 年から 2012 年にかけての JPIX のトラヒックの変動

また、28 日分のトラヒックを特異値分解して得られた特徴成分  $S$  のうち、予測に用いたのは第 1 成分と第 2 成分のみである。この理由として、以下の 2 つの理由が挙げられる。

- 特徴成分  $S$  の内、第 1 成分と第 2 成分のみで元信号の特徴の 90% 以上を表せるから。
- 多くの特徴成分を用いるほど、計算量が多くなるから。

1 つ目の理由についてであるが、図 3.58 より第 1 成分と第 2 成分のみで元信号の特徴を 90% 以上表せる事がわかる。第 3 成分以降は、JPIX のトラヒックの主要な成分を表しているとは言い難いので、本研究では用いない事とした。

2 つ目の理由についてであるが、予測に用いる特徴成分が多くなるほど予測データ作成の為の計算量が多くなる。その為、できるだけ不要な特徴成分は用いない方針とし、本研究では第 1 成分と第 2 成分のみで予測データを作成する事とする。

#### 4.1.2 予測手法

予測手法は、特異値分解における元信号の復元に近い操作で実現する。特異値分解における信号の復元操作を、分かりやすく説明した物を図 4.2, 図 4.3 に示す。

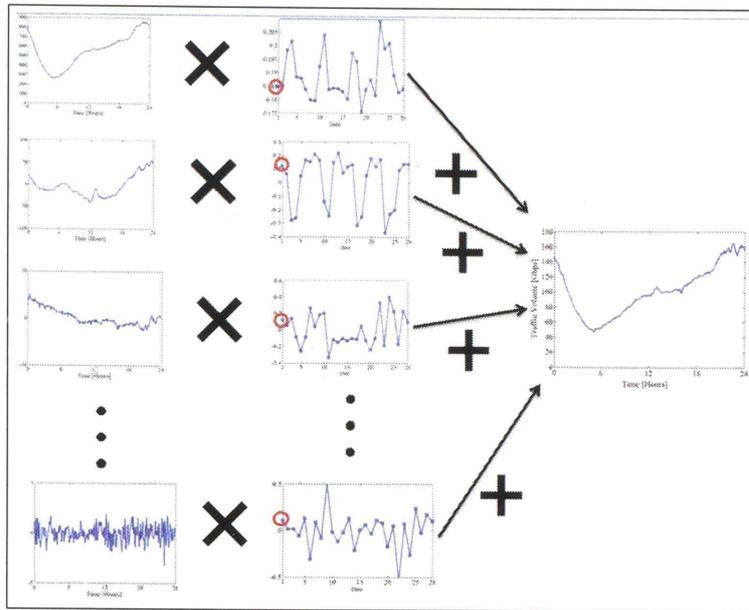


図 4.2: 1日目のトラヒックを復元する場合

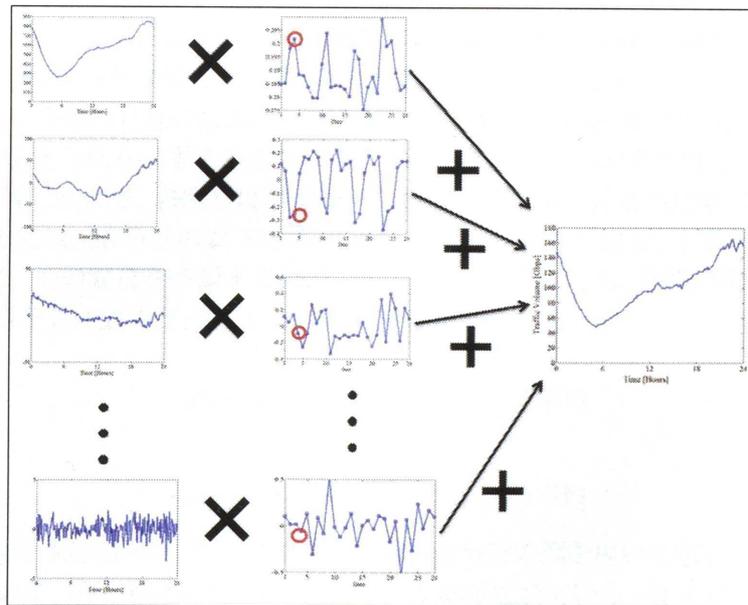


図 4.3: 4日目のトラヒックを復元する場合

図 4.2 は、特異値分解されたトラヒックの内、1日目のトラヒックを復元する例である。図 4.3 は、特異値分解されたトラヒックの内、4日目のトラヒックを復元する例である。第 1 成分～第 28 成分までの各特徴成分の大きさ  $U$  を指定する事で、元トラヒックの復元が可能である。

近い将来における特徴成分  $S'$  が、現在の特徴成分  $S$  に十分近いと仮定できる場合、つまり  $S' = S$  とすれば、予測したい日における特徴成分の大きさ  $U'$  を求める事で予測トラヒック

の作成が可能である。予測に用いる特徴成分は第1成分と第2成分のみなので、第1成分に対する  $U'$  の大きさ  $u'_1$  と、第2成分に対する  $U'$  の大きさ  $u'_2$  を求める事で、予測データの作成が可能である。この事を分かりやすく表すと、図4.4のようになる。

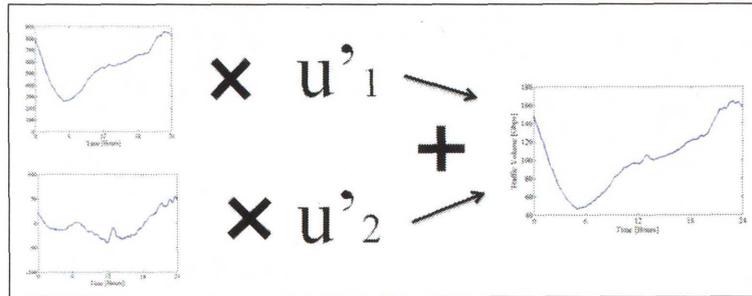


図 4.4: 予測法

この時の、 $u'_1$  と  $u'_2$  は  $U$  から算出する。

#### 4.1.3 $U'$ の算出法

この章では、4.1.2章で説明した  $u'_1$  と  $u'_2$  の算出法について解説する。 $u'_1$  と  $u'_2$  の算出法の基本的な考え方としては以下ようになる。「月曜・火曜・水曜・木曜・金曜・土曜・日曜はおのこの独自の特徴を持っており、その特徴は近い将来のトラヒックにおいても観測される。」つまり、今週の月曜日におけるトラヒックと来週の月曜日におけるトラヒックには、共通した特徴が見られるはずであるという考え方である。本研究では28日分のトラヒックを特異値分解したので、各曜日は丁度4日分ずつ含まれる。将来の月曜日における  $u'_1$  と  $u'_2$  を算出するには、28日分の  $U$  のうち月曜日の  $U$  だけを取り出し、平均を取る事で  $u'_1$  と  $u'_2$  を算出する。他の曜日を予測する場合の  $u'_1$  と  $u'_2$  についても、同様の操作で求める事ができる。この事を式に表すと、式(4.1)、式(4.2)のようになる。

$$u'_1 = \text{同じ曜日における } u_1 \text{ の平均} \quad (4.1)$$

$$u'_2 = \text{同じ曜日における } u_2 \text{ の平均} \quad (4.2)$$

木曜日の  $u'_1$  を求める例を、第3章で示した特異値分解の例を用いて示す。第3章の特異値分解の例において、第1成分に対する  $U$  は図3.57に示されている。図3.57から木曜日の  $u_1$  だけを取り出し、平均を取る事を分かり易く示すと、図4.5のようになる。また、図3.58から木曜日の  $u_2$  だけを取り出し、平均を取る事を分かりやすく示すと、図4.6のようになる。

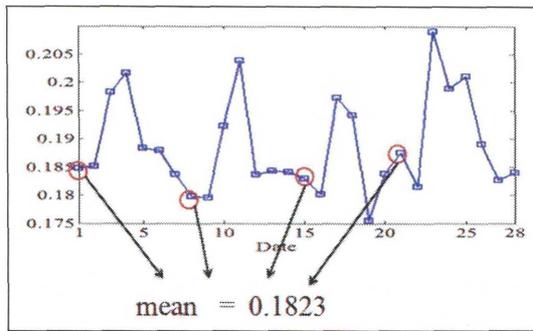


図 4.5: 木曜日の  $u'_1$  の算出例

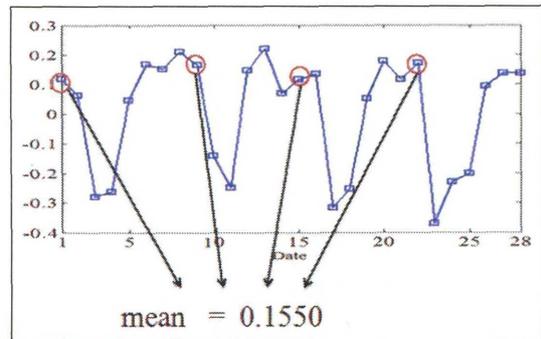


図 4.6: 木曜日の  $u'_2$  の算出例

特異値分解の例において、木曜日のトラヒックは1日目・8日目・15日目・22日目である。その為、図 4.5, 図 4.6 においては1日目, 8日目, 15日目, 22日目の平均を取っている。また、図 4.5, 図 4.6 において平均を取った4点は、いずれも近い値になっている。これは、木曜日には木曜日独自の特徴が毎回観測されるという事を表しており、本研究における予測の前提条件を満たすようになってきている事も分かる。木曜日における  $u'_1 = 0.1823$ 、木曜日における  $u'_2 = 0.1550$  が求められた。

ここで、 $u'_1$  と  $u'_2$  を求める例をもう1例挙げたいと思う。今度は、第3章の特異値分解の例において日曜日における  $u'_1$  と  $u'_2$  を求める。図 3.57 から日曜日の  $u_1$  だけを取り出し、平均を取ったものを図 4.7 に示す。また、図 3.58 から日曜日の  $u_1$  だけを取りだし、平均を取ったものを図 4.8 に示す。

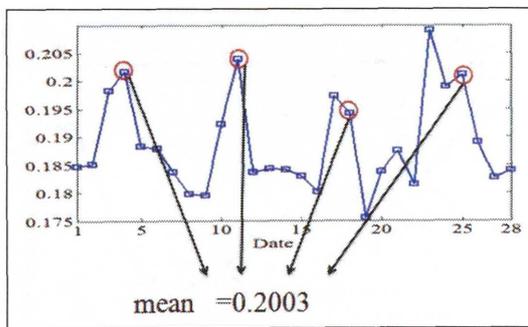


図 4.7: 日曜日の  $u'_1$  の算出例

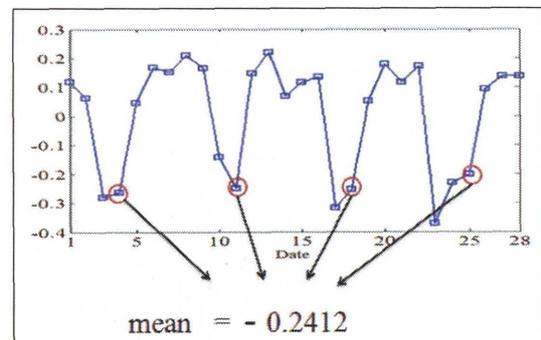


図 4.8: 日曜日の  $u'_2$  の算出例

特異値分解の例において、日曜日のトラヒックは4日目・11日目・18日目・25日目である。その為、図 4.7, 図 4.8 においては4日目, 11日目, 18日目, 25日目の平均を取っている。また、図 4.7, 図 4.8 において平均を取った4点についても、全て近い値になっており本予測法の前提条件を満たしていると分かる。日曜日における  $u'_1 = 0.2003$ 、日曜日における  $u'_2 = -0.2412$  が求められた。

## 4.2 予測結果

この章では、第4.1章で説明した予測法を用いて予測トラフィックを作成し、実際に観測されたトラフィックと比較する事により予測の評価を行う。

### 4.2.1 予測トラフィックの作成

第3章の特異値分解の例を用いて、予測トラフィックの作成を行う。特異値分解によって得られた特徴成分  $S$  のうち、第1成分と第2成分のみを用いて予測トラフィックの作成を行う。予測トラフィックは以下の式(4.3)によって求められる。

$$\text{予測トラフィック} = \text{第1成分} \times u'_1 + \text{第2成分} \times u'_2 \quad (4.3)$$

予測の前提条件を以下に示す。

- 2012年11月1日(木)～2012年11月28日(水)のトラフィックを特異値分解して得られる特徴成分を基礎として予測を行う。
- 予測トラフィックの作成法は、式(4.3)に示した通りである。
- 予測トラフィックを作成する期間は2012年11月29日(木)～2012年12月5日(水)の1週間分である。

以下の図4.9～図4.15に、予測トラフィックと実際のトラフィックを重ねて示す。

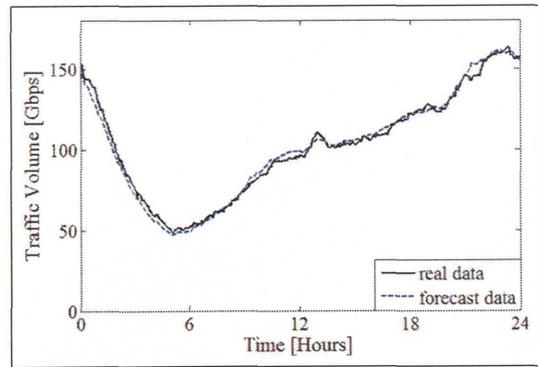
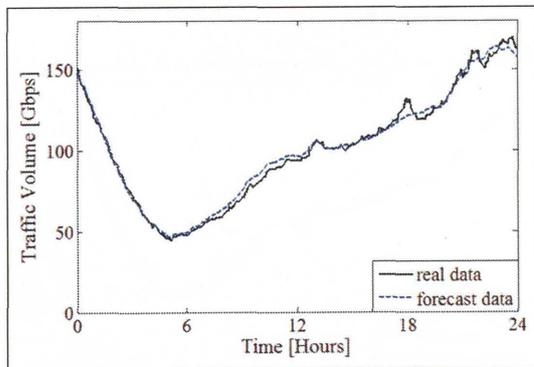


図 4.9: 2012年11月29日(木)の予測結果      図 4.10: 2012年11月30日(金)の予測結果

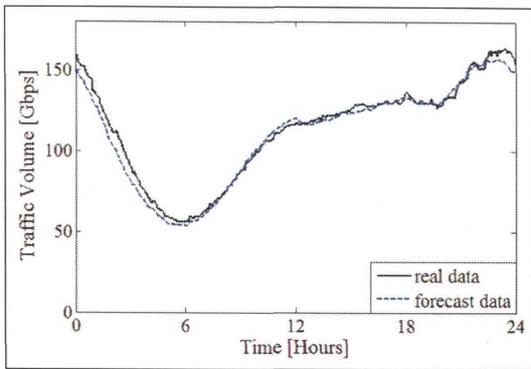


図 4.11: 2012 年 12 月 1 日 (土) の予測結果

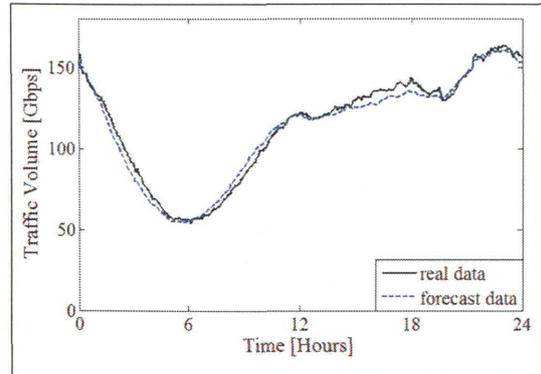


図 4.12: 2012 年 12 月 2 日 (日) の予測結果

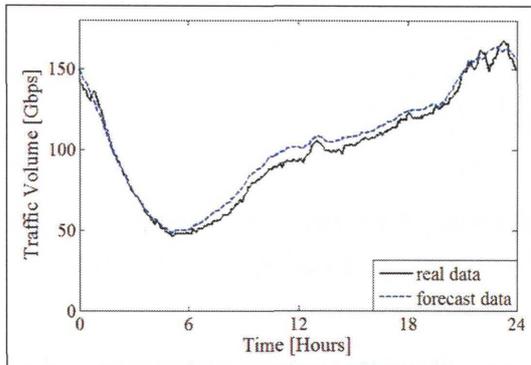


図 4.13: 2012 年 12 月 3 日 (月) の予測結果

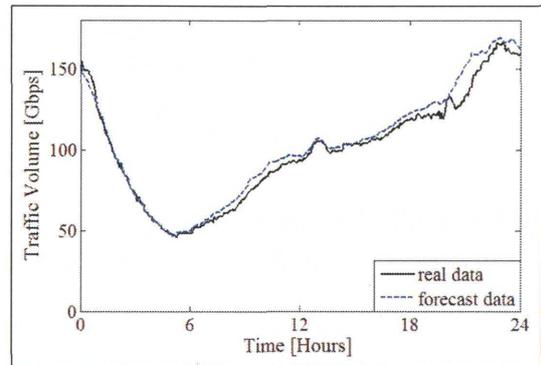


図 4.14: 2012 年 12 月 4 日 (火) の予測結果

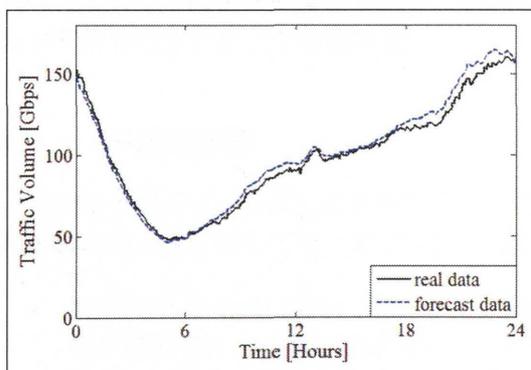


図 4.15: 2012 年 12 月 5 日 (水) の予測結果

図 4.9～図 4.15 を見ると、全ての日において良好な予測が出来ている事が分かる。予測結果を定量的に評価する為に、予測トラフィックと実トラフィックの相対誤差を求める。相対誤差の式を式 (4.4) に示す。

$$\frac{\sum_{i=1}^j \sqrt{(x_i - x'_i)^2}}{\sum_{i=1}^j x_i} \times 100 \text{ [%]} \quad (4.4)$$

ただし、式(4.4)において  $x_i$  は実トラヒックの  $i$  点目を、 $x'_i$  は予測トラヒックの  $i$  点目をそれぞれ表している。また、本研究では1日のトラヒックは287点分なので、 $j$  は287点となる。予測トラヒックと実トラヒックの相対誤差を求めたものを表4.1に示す。

表 4.1: 相対誤差の表

日付	相対誤差 [%]
2012年11月29日(木)	2.4141
2012年11月30日(金)	2.5226
2012年12月1日(土)	2.9479
2012年12月2日(日)	2.8347
2012年12月3日(月)	4.3313
2012年12月4日(火)	3.9526
2012年12月5日(水)	3.8607

表4.1を見ると、どれも相対誤差が5%以内となっており良好な予測が出来ていると分かる。この事より、この例では上手くトラヒックが予測できていたと分かる。

### 4.3 予測結果の統計的性質

第4.2章で1週間分の予測結果について見たが、優れた予測法であるかどうかを判断するにはサンプルが少なすぎる。そこで多くのデータに対して同様の予測を行い、予測結果の統計を見る事で予測手法の評価を行う。今回は、2012年3月10日から2012年12月11日までの276日分のトラヒックを用いた。

#### 4.3.1 予測の手順

今回の予測では、28日分のトラヒックを用いて1日分のトラヒックを予測するという操作を多数のデータに対して行った。つまり、以下に示すような操作を行った。

- 2012年3月11日(日)~2012年4月7日(土)までのトラヒックを用いて、4月8日(日)の予測トラヒックを作成する。
- 2012年3月12日(月)~2012年4月8日(日)までのトラヒックを用いて、4月9日(月)の予測トラヒックを作成する。
- 2012年3月13日(火)~2012年4月9日(月)までのトラヒックを用いて、4月10日(火)の予測トラヒックを作成する。
- 途中省略
- 2012年11月13日(火)~2012年12月10日(月)までのトラヒックを用いて、12月11日(火)の予測トラヒックを作成する。

この様な操作を行う事により, 2012年4月8日(日)から2012年12月11日(火)までの248日分の予測トラヒックが得られる. このようにして得られた多数の予測トラヒックから, 予測トラヒックの統計的性質を明らかにする.

ここでの, 予測トラヒックの作成法としては, 第4.1章で示した予測法を用いた. 簡単に説明すると, 28日分のトラヒックを特異値分解して得られた特徴成分  $S$  に対して,  $U$  から算出した  $U'$  を掛け合わせることで, 予測トラヒックを作成した. なお, 予測に用いた特徴成分は第1成分と第2成分だけである.

予測トラヒックと, 実際のトラヒックとの相対誤差を表4.2~4.10に示す.

表 4.2: 4月における相対誤差の表

日付	相対誤差 [%]	日付	相対誤差 [%]	日付	相対誤差 [%]
4月8日(土)	5.84	4月16日(日)	5.75	4月24日(月)	9.56
4月9日(日)	6.22	4月17日(月)	13.57	4月25日(火)	6.52
4月10日(月)	14.96	4月18日(火)	8.49	4月26日(水)	3.07
4月11日(火)	3.70	4月19日(水)	6.64	4月27日(木)	4.41
4月12日(水)	6.33	4月20日(木)	6.55	4月28日(金)	5.95
4月13日(木)	7.12	4月21日(金)	4.19	4月29日(土)	6.86
4月14日(金)	2.43	4月22日(土)	4.19	4月30日(日)	11.67
4月15日(土)	2.92	4月23日(日)	4.18		

表 4.3: 5月における相対誤差の表

日付	相対誤差 [%]	日付	相対誤差 [%]	日付	相対誤差 [%]
5月1日(火)	3.98	5月12日(土)	4.54	5月23日(水)	4.68
5月2日(水)	8.78	5月13日(日)	1.77	5月24日(木)	8.19
5月3日(木)	15.66	5月14日(月)	3.71	5月25日(金)	3.93
5月4日(金)	12.27	5月15日(火)	6.29	5月26日(土)	2.63
5月5日(土)	5.47	5月16日(水)	1.77	5月27日(日)	1.99
5月6日(日)	3.73	5月17日(木)	4.12	5月28日(月)	6.34
5月7日(月)	4.02	5月18日(金)	3.91	5月29日(火)	5.93
5月8日(火)	2.04	5月19日(土)	1.51	5月30日(水)	7.40
5月9日(水)	3.07	5月20日(日)	1.38	5月31日(木)	7.63
5月10日(木)	2.64	5月21日(月)	4.50		
5月11日(金)	2.31	5月22日(火)	1.92		

表 4.4: 6月における相対誤差の表

日付	相対誤差 [%]	日付	相対誤差 [%]	日付	相対誤差 [%]
6月1日(金)	4.95	6月11日(月)	2.05	6月21日(木)	4.38
6月2日(土)	2.19	6月12日(火)	3.15	6月22日(金)	2.23
6月3日(日)	1.81	6月13日(水)	3.41	6月23日(土)	1.83
6月4日(月)	2.05	6月14日(木)	1.94	6月24日(日)	1.68
6月5日(火)	2.36	6月15日(金)	2.00	6月25日(月)	1.38
6月6日(水)	2.43	6月16日(土)	6.20	6月26日(火)	3.71
6月7日(木)	1.78	6月17日(日)	2.21	6月27日(水)	1.39
6月8日(金)	2.28	6月18日(月)	2.91	6月28日(木)	2.56
6月9日(土)	3.98	6月19日(火)	10.88	6月29日(金)	2.32
6月10日(日)	2.14	6月20日(水)	1.65	6月30日(土)	2.16

表 4.5: 7月における相対誤差の表

日付	相対誤差 [%]	日付	相対誤差 [%]	日付	相対誤差 [%]
7月1日(日)	6.59	7月12日(木)	2.60	7月23日(月)	3.26
7月2日(月)	2.37	7月13日(金)	1.76	7月24日(火)	3.45
7月3日(火)	2.28	7月14日(土)	4.70	7月25日(水)	3.32
7月4日(水)	2.60	7月15日(日)	6.95	7月26日(木)	6.55
7月5日(木)	1.48	7月16日(月)	10.39	7月27日(金)	5.46
7月6日(金)	1.83	7月17日(火)	5.63	7月28日(土)	3.95
7月7日(土)	1.98	7月18日(水)	1.76	7月29日(日)	2.61
7月8日(日)	1.97	7月19日(木)	1.48	7月30日(月)	3.10
7月9日(月)	2.35	7月20日(金)	2.90	7月31日(土)	4.50
7月10日(火)	5.76	7月21日(土)	2.47		
7月11日(水)	2.08	7月22日(日)	3.89		

表 4.6: 8月における相対誤差の表

日付	相対誤差 [%]	日付	相対誤差 [%]	日付	相対誤差 [%]
8月1日(水)	3.80	8月12日(日)	6.11	8月23日(木)	2.54
8月2日(木)	3.58	8月13日(月)	4.52	8月24日(金)	2.11
8月3日(金)	3.59	8月14日(火)	8.95	8月25日(土)	1.87
8月4日(土)	5.48	8月15日(水)	9.82	8月26日(日)	2.63
8月5日(日)	2.40	8月16日(木)	9.19	8月27日(月)	1.94
8月6日(月)	3.23	8月17日(金)	7.98	8月28日(火)	2.97
8月7日(火)	3.95	8月18日(土)	5.15	8月29日(水)	3.94
8月8日(水)	2.92	8月19日(日)	7.22	8月30日(木)	2.18
8月9日(木)	2.14	8月20日(月)	4.93	8月31日(金)	2.77
8月10日(金)	3.80	8月21日(火)	2.80		
8月11日(土)	3.05	8月22日(水)	2.18		

表 4.7: 9月における相対誤差の表

日付	相対誤差 [%]	日付	相対誤差 [%]	日付	相対誤差 [%]
9月1日(土)	4.23	9月11日(火)	6.63	9月21日(金)	3.25
9月2日(日)	3.90	9月12日(水)	6.04	9月22日(土)	4.45
9月3日(月)	5.39	9月13日(木)	5.64	9月23日(日)	7.08
9月4日(火)	7.65	9月14日(金)	5.08	9月24日(月)	5.94
9月5日(水)	7.08	9月15日(土)	2.33	9月25日(火)	3.70
9月6日(木)	6.91	9月16日(日)	1.54	9月26日(水)	4.42
9月7日(金)	6.71	9月17日(月)	11.49	9月27日(木)	4.23
9月8日(土)	2.94	9月18日(火)	1.58	9月28日(金)	2.89
9月9日(日)	1.96	9月19日(水)	2.67	9月29日(土)	6.43
9月10日(月)	5.89	9月20日(木)	2.97	9月30日(日)	14.01

表 4.8: 10月における相対誤差の表

日付	相対誤差 [%]	日付	相対誤差 [%]	日付	相対誤差 [%]
10月1日(月)	6.51	10月12日(金)	12.68	10月23日(火)	4.43
10月2日(火)	5.83	10月13日(土)	9.71	10月24日(水)	2.47
10月3日(水)	7.82	10月14日(日)	14.79	10月25日(木)	2.34
10月4日(木)	10.38	10月15日(月)	14.35	10月26日(金)	1.80
10月5日(金)	10.97	10月16日(火)	7.52	10月27日(土)	3.32
10月6日(土)	10.42	10月17日(水)	6.32	10月28日(日)	1.94
10月7日(日)	16.27	10月18日(木)	5.12	10月29日(月)	2.57
10月8日(月)	6.55	10月19日(金)	2.54	10月30日(火)	2.31
10月9日(火)	5.78	10月20日(土)	5.54	10月31日(水)	2.41
10月10日(水)	6.44	10月21日(日)	8.18		
10月11日(木)	7.70	10月22日(月)	5.41		

表 4.9: 11月における相対誤差の表

日付	相対誤差 [%]	日付	相対誤差 [%]	日付	相対誤差 [%]
11月1日(木)	3.27	11月11日(日)	3.42	11月21日(水)	2.33
11月2日(金)	4.98	11月12日(月)	2.01	11月22日(木)	2.78
11月3日(土)	2.26	11月13日(火)	2.64	11月23日(金)	14.71
11月4日(日)	2.30	11月14日(水)	2.14	11月24日(土)	2.92
11月5日(月)	2.41	11月15日(木)	1.84	11月25日(日)	3.42
11月6日(火)	3.07	11月16日(金)	2.66	11月26日(月)	3.61
11月7日(水)	2.61	11月17日(土)	3.16	11月27日(火)	2.88
11月8日(木)	2.58	11月18日(日)	5.50	11月28日(水)	2.18
11月9日(金)	1.85	11月19日(月)	6.44	11月29日(木)	2.41
11月10日(土)	3.64	11月20日(火)	2.65	11月30日(金)	3.35

表 4.10: 12月における相対誤差の表

日付	相対誤差 [%]	日付	相対誤差 [%]	日付	相対誤差 [%]
12月1日(土)	3.36	12月5日(水)	3.99	12月9日(日)	3.88
12月2日(日)	2.95	12月6日(木)	2.39	12月10日(月)	4.29
12月3日(月)	2.56	12月7日(金)	7.47	12月11日(火)	2.95
12月4日(火)	2.74	12月8日(土)	2.08		

表 4.2~4.10 を見ると、相対誤差が異常に高い日付があることが分かる。相対誤差が高くなっている日付として、例えば以下の日付がある。

- 2012年4月30日(月曜日)
- 2012年5月3日(木曜日)
- 2012年5月4日(金曜日)
- 2012年7月16日(月曜日)
- 2012年9月17日(月曜日)
- 2012年11月23日(金曜日)

ここで、2012年4月8日~12月11日までにおける祝日を以下に挙げる。

- 2012年4月29日(日曜日)：昭和の日
- 2012年4月30日(月曜日)：振替休日 \*
- 2012年5月3日(木曜日)：憲法記念日 \*
- 2012年5月4日(金曜日)：みどりの日 \*
- 2012年5月5日(土曜日)：こどもの日
- 2012年7月16日(月曜日)：海の日 \*
- 2012年9月17日(月曜日)：敬老の日 \*
- 2012年9月22日(土曜日)：秋分の日
- 2012年10月8日(月曜日)：体育の日
- 2012年11月3日(土曜日)：文化の日
- 2012年11月23日(金曜日)：勤労感謝の日 \*

祝日の内、「\*」で示した部分は相対誤差が高い日付と一致している。このように、祝日において高い相対誤差が観測されるのには理由がある。祝日には平日とは違ったトラヒックが観測され、その特徴は土曜日・日曜日に近い特徴を持っている。図 4.16 に、2012年11月23日(金)の勤労感謝の日におけるトラヒックのグラフを示す。

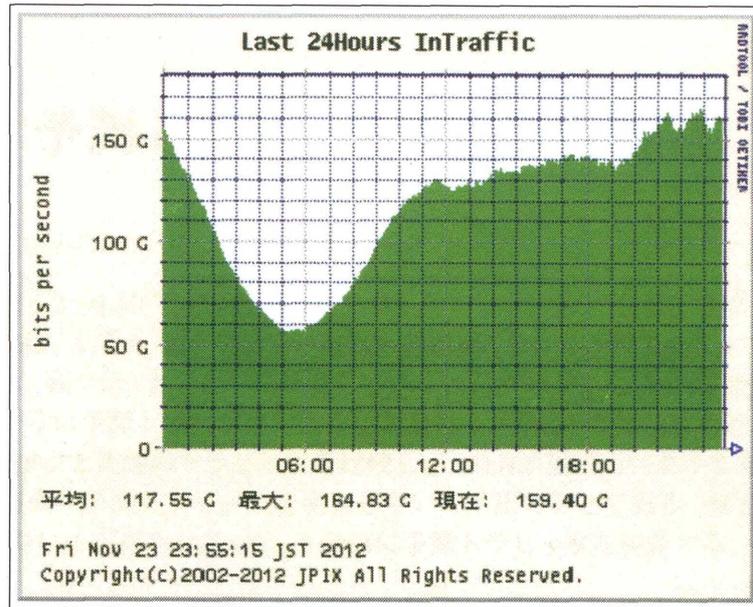


図 4.16: 2012年11月23日(金) 勤労感謝の日におけるトラヒック

図 4.16 を見ると、12時～19時にかけて平日より高いトラヒックが観測されており、土曜日・日曜日のトラヒックの特徴に近いことが分かる。祝日には、土曜日・日曜日に近い特徴が観測される為、平日であると仮定して作成された予測トラヒックと比較した時に、相対誤差が高くなってしまった。

逆に、土曜日・日曜日における祝日においては、相対誤差がそれほど高くなっていない事が分かる。これは、土曜日・日曜日と仮定して作成した予測トラヒックの特徴が、祝日におけるトラヒックの特徴と近い為、相対誤差がそれほど大きくならなかった為である。

以下に示す表 4.11 に、248 日分の相対誤差の最大、最小、平均、中央値、分散値を示す。

表 4.11: 12月における相対誤差の表

	相対誤差 [%]
248 日分における最大	16.27
248 日分における最小	1.38
248 日分における平均	4.57
248 日分における中央値	3.60
248 日分における標準偏差	3.02

表 4.11 より、平均相対誤差 4.57% で標準偏差 3.02% の予測ができていると分かる。このことより、比較的良好な予測が行えている事がわかる。

以上が予測法とその評価についての解説である。

## 第5章 予測トラヒックの逐次更新

第4.3章の表4.2～4.10で示した通り、予測トラヒックの相対誤差にはばらつきが見られる。言い換えれば、上手く予測できている日もあれば、上手く予測できていない日もあるということである。我々は、上手く予測ができていない日であっても、相対誤差が高くなったままにならないように予測トラヒックの逐次更新を行う。予測トラヒックの逐次更新というのは、予測トラヒックと実際のトラヒックを比較して、相対誤差が高いようであれば、相対誤差が小さくなる方向へ予測トラヒックを更新していく手法のことである。JPIXのトラヒックデータは、5分毎に1点得られるので、5分毎に予測トラヒックを更新する。

第5章では、予測トラヒックの逐次更新法を紹介し、予測トラヒックを更新する事によってどれだけ相対誤差を小さくできるかを評価する。

### 5.1 逐次更新手法

この第5.1章では、予測トラヒックの逐次更新手法を解説する。少し難解な手法である為、できるだけイメージ図を示し分かり易く解説する。予測トラヒックの逐次更新手法の手順を簡単に説明すると、以下ようになる。

1. 予測日の前日までに、予測トラヒックを作成しておく。
2. 予測日当日になり、JPIXのトラヒックが5分毎に1点得られる。
3. 予測トラヒックを元にして、多数の予測トラヒック候補を作成する。
4. 実際に得られたトラヒックと、多数の予測トラヒック候補を比較する。
5. 手順4において、最も相対誤差が小さくなった予測トラヒック候補を、新たな予測トラヒックとして採用。
6. 0時～24時の間、手順2～4を繰り返す。

#### 5.1.1 基本方針

第4.3章で説明した方法を用いて予測トラヒックを作成し、その予測トラヒックに対して逐次更新を行う。予測トラヒックの作成法を簡単に説明すると、28日分の連続したトラヒックを元に、将来1日分の予測トラヒックを作成する。1つ例を挙げると、『2012年3月11日(日)～2012年4月7日(土)までのトラヒックを用いて、4月8日(日)の予測トラヒックを作成する。』といった形である。このように作成した予測トラヒックと、予測日当日に得られた実際のトラヒックを比較する。例えば『あらかじめ用意した2012年4月8日(日)の予測トラヒックと、2012年4月8日(日)に実際に得られたトラヒックを比較して、誤差があるよ

うであれば予測トラフィックを実際に得られたトラフィックに近づける。』という操作を行う。  
この事をわかりやすく説明したものを図 5.1, 図 5.2 に示す。

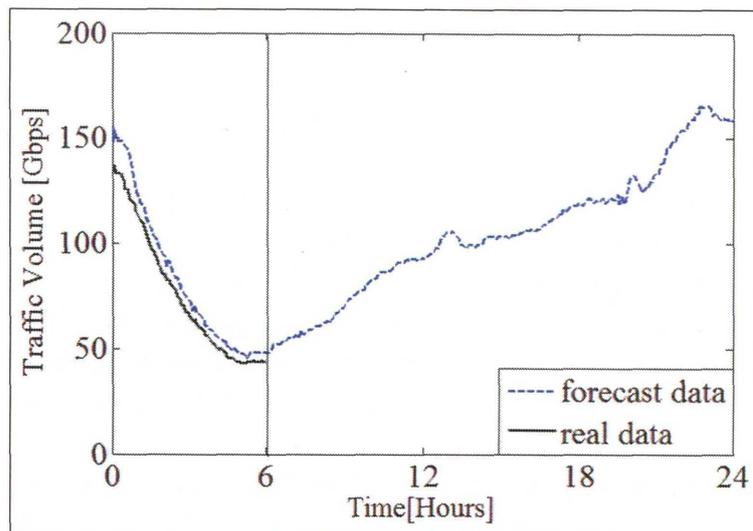


図 5.1: 予測トラフィックと実際のトラフィックの比較

図 5.1 は, 予測日当日に 6 時間分のトラフィックが得られた時の様子を示しており, 点線が予測トラフィック, 実線が実際に得られたトラフィックを示しているとする. 図 5.1 を見ると, 実際に得られたトラフィックは, 予測トラフィックに比べて小さい値を示すようになっている事が分かる. このような場合, 6 時以降のトラフィックについても予測トラフィックに比べて低くなる可能性が高いと予想される. そこで, 予測トラフィックを実際のトラフィックに合わせて低くし, 誤差が小さくなる方向へ更新する. このことを表したものを図 5.2 に示す.

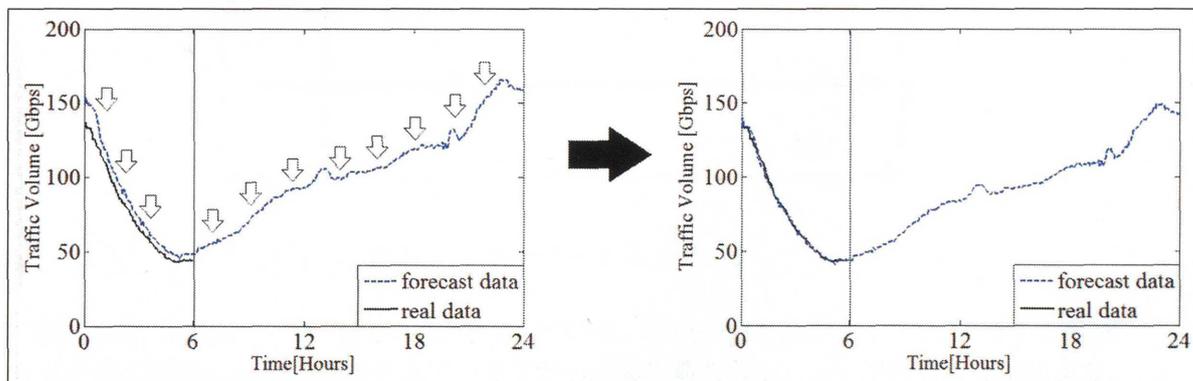


図 5.2: 予測トラフィックの更新

図 5.2 のような更新を行う事で, より妥当な予測トラフィックになったと言える. このように逐次更新を行う事で, 予測トラフィックの妥当性を高めていく. 本研究における逐次更新法は, 単なる予測トラフィックのボリューム増減ではなく, 各特徴成分をそれぞれ増減する事によって実現している. その手法は, 第 5.1.2 章以降で説明する.

### 5.1.2 逐次更新手法

予測トラヒックは、28日分のトラヒックを特異値分解して得られた特徴成分のうち、第1成分と第2成分のみを用いて作成している。第4.2章の式(4.3)にて、予測トラヒックの式を示したが、もう一度同じものを式(5.1)に示す。

$$\text{予測トラヒック} = \text{第1成分} \times u'_1 + \text{第2成分} \times u'_2 \quad (5.1)$$

式(5.1)において、 $u'_1$ は第1成分の大きさを、 $u'_2$ は第2成分の大きさを示している。この事からも分かるように、予測トラヒックは第1成分と第2成分の組み合わせによって作成されており、 $u'_1$ を増減させれば予測トラヒックに含まれる第1成分が増減し、 $u'_2$ を増減させれば予測トラヒックに含まれる第2成分が増減する。このようにして、予測トラヒックに含まれる第1成分・第2成分を増減させ、実際に観測されたトラヒックに最も上手く一致したものを新たな予測トラヒックとして採用する。

増減させるのは、 $u'_1$ 、 $u'_2$ の2つの変数であるので、2次元サーチを行えば良いと分かる。 $u'_1$ 、 $u'_2$ における2次元サーチのイメージを、図5.3に示す。

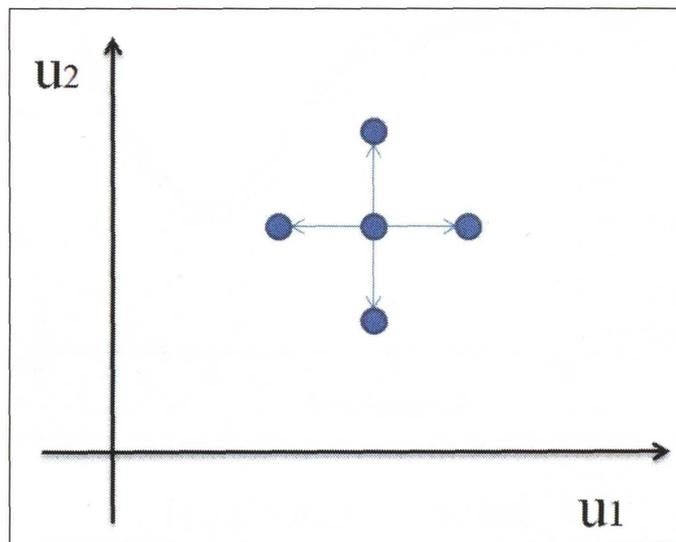


図 5.3: 2次元サーチのイメージ

図5.3は、2次元サーチのイメージを示している。中央の点を基準として、 $u'_1$ 、 $u'_2$ を増減させる事で新たに4つの点ができている事が分かる。新たな $u'_1$ 、 $u'_2$ が得られた事により、中央の点も含め、5つの予測トラヒック候補が出来上がる。トラヒック候補を表すと、式(5.2)～式(5.7)になる。

$$\text{予測トラヒック候補 1(上の点)} = \text{第1成分} \times u'_1 + \text{第2成分} \times (u'_2 + \beta) \quad (5.2)$$

$$\text{予測トラヒック候補 2(左の点)} = \text{第1成分} \times (u'_1 - \alpha) + \text{第2成分} \times u'_2 \quad (5.3)$$

$$\text{予測トラヒック候補 3(中央の点)} = \text{第1成分} \times u'_1 + \text{第2成分} \times u'_2 \quad (5.4)$$

$$\text{予測トラヒック候補 4(右の点)} = \text{第1成分} \times (u'_1 + \alpha) + \text{第2成分} \times u'_2 \quad (5.5)$$

$$\text{予測トラヒック候補 5(下の点)} = \text{第1成分} \times u'_1 + \text{第2成分} \times (u'_2 - \beta) \quad (5.6)$$

式(5.2)～式(5.6)において、 $\alpha$ は $u'_1$ の増減分を、 $\beta$ は $u'_2$ の増減分をそれぞれ表している。このようにして $u'_1$ 、 $u'_2$ を増減させ5つの予測トラヒックを作成し、重ねて表示したものを図5.4に示す。

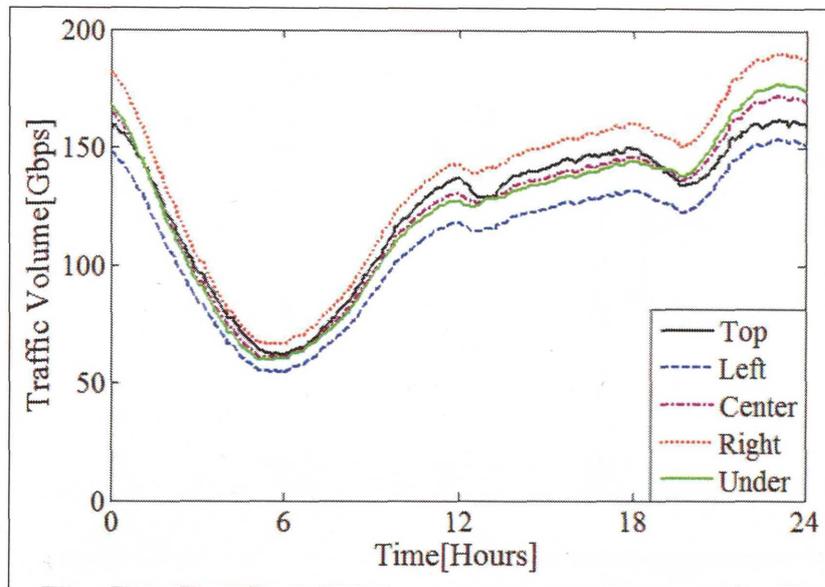


図 5.4: 予測トラヒックの候補

図5.4を見ると、 $u'_1$ 、 $u'_2$ の増減によって5つの予測トラヒック候補ができていると分かる。また、単なるボリュームの増減による作成と違い、一つ一つ波形の特徴が異なっていると分かる。これによって、単なるボリュームの増減を行うときより精密に、予測トラヒックを実際のトラヒックに合わせて込む事が可能である。今回は分かり易い例として示すため、 $u'_1$ 、 $u'_2$ を大幅に変動させたが、実際に逐次更新を行う時にはもっと小さい範囲で $u'_1$ 、 $u'_2$ を変動させ、少しずつ実際のトラヒックと合わせていく。そして、最も実際のトラヒックに近いもの(相対誤差が小さくなったもの)を次の予測トラヒックとするのである。JPIXのトラヒックデータは、5分毎に1点新たなデータが得られるので、5分毎にこの操作を繰り返す事になる。以上が逐次更新法の概要である。詳しい方法については第5.1.3章以降で示す。

### 5.1.3 実際の2次元サーチ

2次元サーチの概要については、第5.1.2章で示した通りであるが、ここでは実際の方法を説明していく。我々が用いた2次元サーチの方法は、一般的に『ランダムサーチ』や『山登り法』と呼ばれる手法である(以降ランダムサーチと呼ぶ)。

簡単に説明すると、現在の基準点の近傍を複数箇所探索し、最も優れている箇所を次の基準点とする。この操作を繰り返していき、現在の基準点が最も優れていると判定されたとき、現在の基準点はその近傍のなかで最も優れた点であると言える。このようにして、優れた点を探すのがランダムサーチである。我々は、 $u_1, u_2$ の2次元空間に対してこのランダムサーチを用い、我々の方法では、近傍を四角形のように探索する方法を用いた。このことを図で示したものを、図5.5に示す。

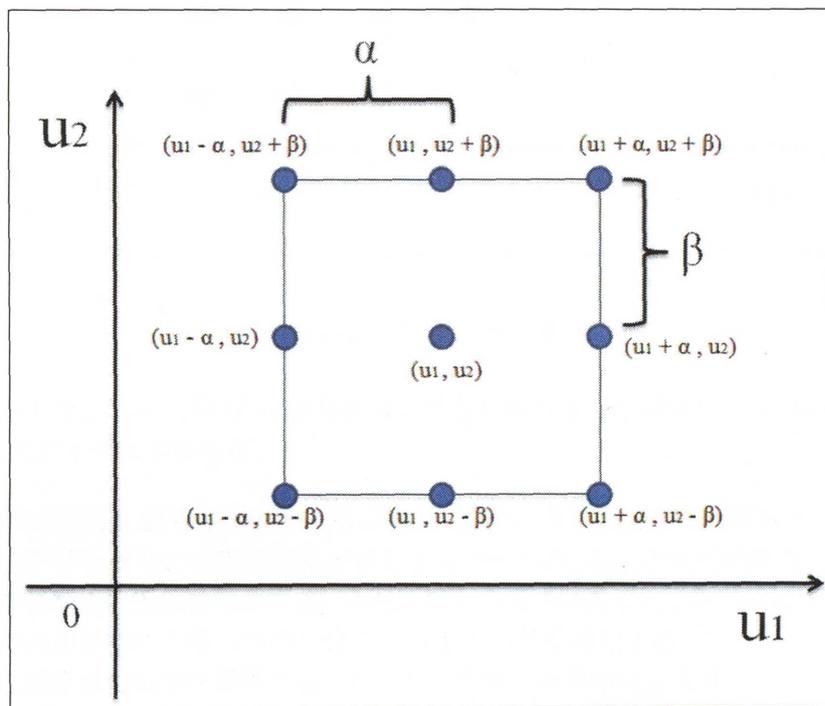


図 5.5: 近傍探索

図5.5において、 $\alpha$ は $u_1$ の1ステップあたりの変動分を、 $\beta$ は $u_2$ の1ステップあたりの変動分を表している( $\alpha, \beta$ の求め方は後で説明する)。この探索法では、中央の $(u_1, u_2)$ を現在の基準点として、図5.5のように近傍の8点を探索する。各点に対して、1つの予測トラヒックが作成できるので、中央の点を含め9個の予測トラヒックを作成する。9個の予測トラヒックと、実際に観測されたトラヒックを比較して、最も相対誤差が小さくなった予測トラヒックと対応する点を次の基準点とする。『中央の点から作成された予測トラヒックが最も相対誤差が小さい』と判定されるまで、この操作を何度も繰り返し、中央が最も優れていると判定されれば、 $\alpha, \beta$ をそれぞれ $\frac{1}{2}$ 倍して、さらに精密に近傍を探索する。この事を表したものを図5.6に示す。

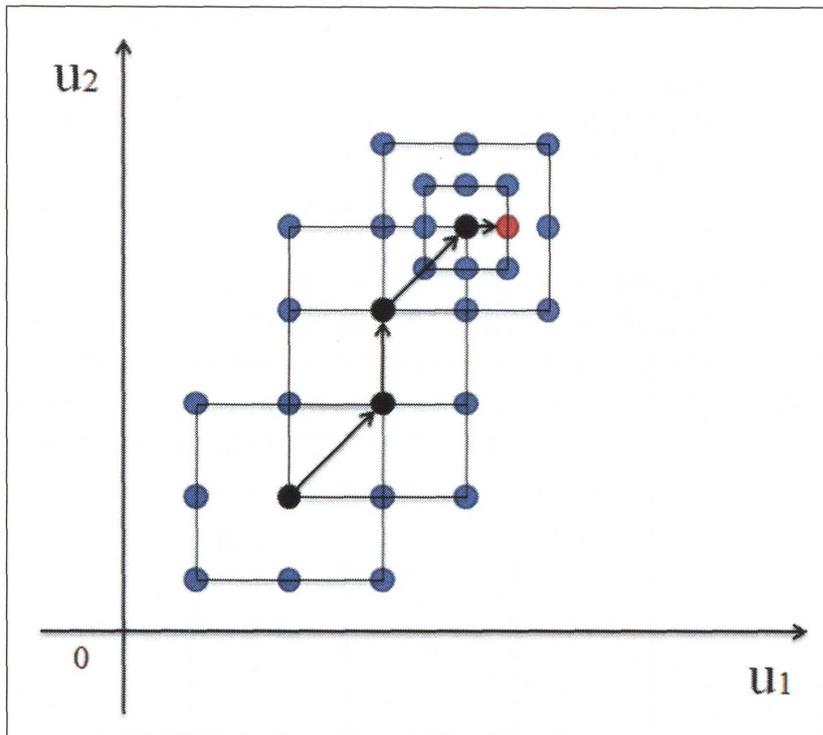


図 5.6: ランダムサーチ

図 5.6 に示したように、四角形による探索を何度も繰り返して探索を行う。図 5.6 の図を順に説明すると以下ようになる。

- 1 回目の探索は、最も左下の四角形である。右上の点で作成した予測トラヒック候補が、実際のトラヒックとの相対誤差が最も小さかった為、右上を次の基準点にする。
- 新たな基準点の近傍を探索する。今回は真上の点で作成した予測トラヒック候補が、最も相対誤差が小さくなったので真上の点を次の基準点とする。
- 新たな基準点の近傍を探索する。右上の点を新たな基準点とする。
- 新たな基準点の近傍を探索する。今回は中央の点で作成した予測トラヒック候補が、最も相対誤差が小さくなったので探索範囲  $\alpha$ ,  $\beta$  をそれぞれ  $\frac{1}{2}$  倍する。
- $\frac{1}{2}$  倍された  $\alpha$ ,  $\beta$  の範囲で探索する。今回は右の点で作成した予測トラヒック候補が、最も相対誤差が小さくなったので右の点を次の基準点とする。
- 新たな基準点の近傍を探索する。今回は中央の点で作成した予測トラヒック候補が、最も相対誤差が小さくなった。探索範囲  $\alpha$ ,  $\beta$  は既に十分小さいので、精密な合わせ込みができたとして探索を終了。
- 現在の基準点を元にして作成した予測トラヒック候補を、予測トラヒックとして採用する。

以上が実際に用いている 2 次元サーチ法である。実際の 2 次元サーチでは、 $\alpha$ ,  $\beta$  をそれぞれ  $\frac{1}{2}$  倍する操作を 3 回行い、その後選ばれた中央の点を元にして作成した予測トラヒックを採用する。

#### 5.1.4 四角形サーチにおけるアルゴリズム

本研究では、ランダムサーチにおいて四角形で探索を行ったが、実は四角形でサーチを行ったのには理由がある。例えば、基準点の近傍を円形に探索する方法もあるが、四角形に比べて計算量が増えてしまう。この章では、四角形でサーチした理由を述べ、さらに四角形でサーチする際の実際のアルゴリズムの説明も行う。

まず初めに、四角形のサーチと円形サーチとの比較を行う。四角形サーチには、主に『1回目のサーチを行うパターン』、『2回目以降で上下左右の点を選ばれるパターン』、『2回目以降で斜めの点を選ばれるパターン』の3つのパターンがある。円系サーチには、主に『1回目のサーチを行うパターン』、『2回目のサーチを行うパターン』がある。図 5.7～図 5.9 に四角形サーチを、図 5.10, 図 5.11 に円形サーチをそれぞれ示す。

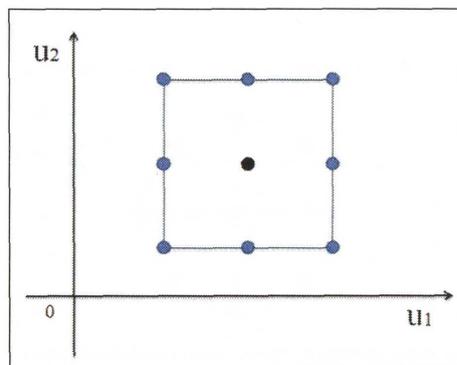


図 5.7: 四角形サーチ (1 回目のサーチ)

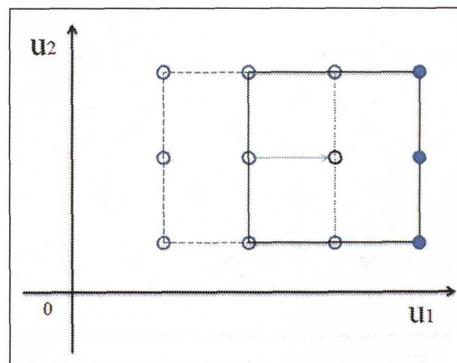


図 5.8: 四角形サーチ (2 回目以降で右の点を選ばれた時)

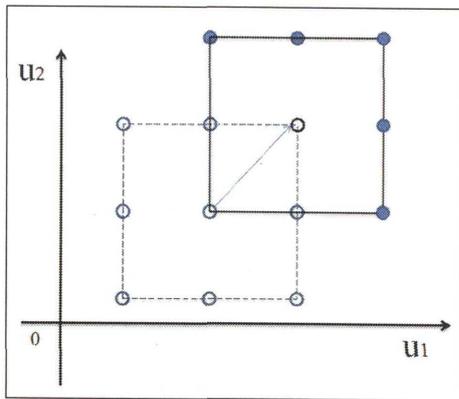


図 5.9: 四角形サーチ (2 回目以降で斜めの点が選ばれた時)

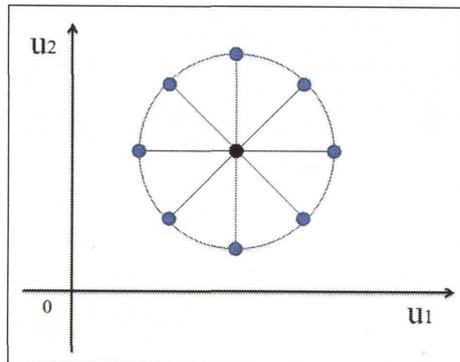


図 5.10: 円形サーチ (1 回目のサーチ)

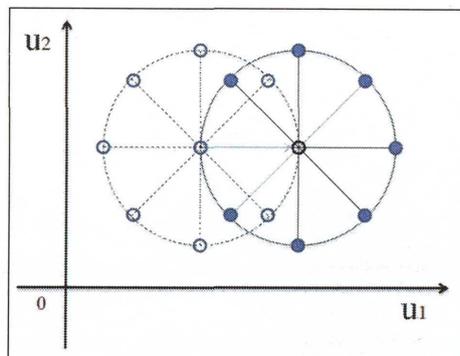


図 5.11: 円形サーチ (2 回目以降のサーチ)

図 5.7 は四角形サーチにおける 1 回目のサーチを表している。図 5.7 において、中央の点が既知であるとする、未知の点は 8 点である。

図 5.8 は四角形サーチにおいて、2 回目以降で右の点が選ばれた時である。図 5.8 において、未知の点は 3 点である。ここでは、右の点が選ばれた例を示しているが、上・下・左の点が選

ばれた時でも同様に未知の点は3点である。

図 5.9 は四角形サーチにおいて、2 回目以降で斜めの点が選ばれた時である。図 5.9 において、未知の点は5点である。ここでは、右斜め上の点が選ばれた例を示しているが、他の斜めの点が選ばれた時でも同様に未知の点は5点である。

図 5.10 は円形サーチにおける1 回目のサーチを表している。図 5.10 において、中央の点が既知であるとする、未知の点は8点である。

図 5.11 は円形サーチにおける2 回目以降のサーチを表している。図 5.11 において未知の点は7点である。ここでは、右の点が選ばれた例を示しているが、どの点が選ばれたとしても、未知の点は7点である。以上の事を表にまとめると、表 5.1 のようになる。

表 5.1: 各サーチ法における未知の点

	四角形サーチ	円形サーチ
1 回目	8 個	8 個
2 回目以降	上下左右 : 3 個 斜め : 5 個	7 個

表 5.1 より、四角形サーチの方が少ない計算量でサーチを行う事ができると分かる。これが、四角形サーチを採用した理由である。

以降、四角形サーチのアルゴリズムについて説明する。四角形サーチの1 回目において、各点に数字を割り振ると図 5.12 のようになる。

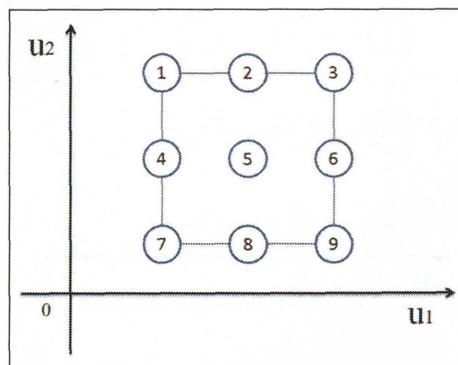


図 5.12: 1 回目のサーチ

この図 5.12 を基本として、1~9 の各点が選ばれる場合を全て網羅する事で、アルゴリズムを実現する事が可能である。図 5.13 に が選ばれたパターンを示す。

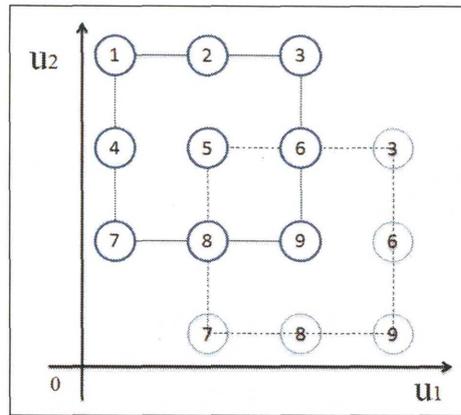


図 5.13: が選ばれたパターン

図 5.13 について説明する。 が選ばれると、以下の操作を行う。

- 現在の を新たな四角の へ
- 現在の を新たな四角の へ
- 現在の を新たな四角の へ
- 現在の を新たな四角の へ
- 新たな四角の 、 、 、 、 は未知の値なので新たに求める。

図 5.14 に が選ばれたパターンを示す。

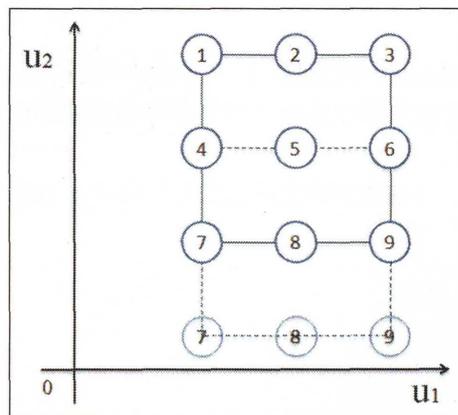


図 5.14: が選ばれたパターン

図 5.14 について説明する。 が選ばれると、以下の操作を行う。

- 現在の を新たな四角の へ

- 現在の  $u_1$  を新たな四角の  $u_1$  へ
- 新たな四角の  $u_2$ 、 $u_3$ 、 $u_4$  は未知の値なので新たに求める。

$u_1$  が選ばれたパターンは以上の通りである。 $u_2$ 、 $u_3$ 、 $u_4$ 、 $u_5$ 、 $u_6$  が選ばれたパターンも同様な操作で実現することが出来るので、ここでは省略させてもらう。次に、 $u_7$  が選ばれたパターンの説明をする。 $u_7$  が選ばれたパターンを図 5.15 に示す。

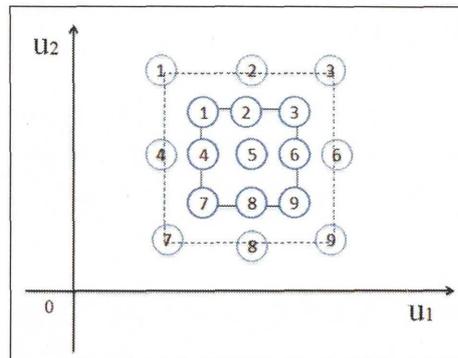


図 5.15:  $u_7$  が選ばれたパターン

図 5.15 について説明する。 $u_7$  が選ばれると、以下の操作を行う。

- 現在の  $u_1$  を新たな四角の  $u_1$  へ
- 探索範囲を、現在の  $\frac{1}{2}$  へ
- 新たな四角の  $u_2$ 、 $u_3$ 、 $u_4$ 、 $u_5$ 、 $u_6$  は未知の値なので新たに求める。

また、本研究におけるランダムサーチは、 $u_7$  が 4 回選択された時点で終了とする。つまり、探索範囲を 3 回狭め、その後中央の点  $u_5$  が選ばれた時点で終了とする。

以上が、四角形サーチにおけるアルゴリズムの説明である。

### 5.1.5 探索範囲 $\alpha, \beta$ の算出法について

この第5.1.5章では、 $u_1$ の1ステップ当たりの探索範囲 $\alpha$ と、 $u_2$ の1ステップ当たりの探索範囲 $\beta$ を求める。本研究では、JPIXのトラヒックを研究対象としている為、JPIXにおける探索範囲を求めているが、今回用いる方法はJPIX以外の任意の信号にも適用できるような方法を取っている。 $\alpha$ と $\beta$ は、任意の信号を特異値分解をして得られた $U$ から算出する事が可能である。 $U$ の分散を見ることで、第1・第2成分をどれだけ変動させるのが妥当であるかを知ることができる。 $U$ の分散が大きい信号であれば、普段から第1・第2成分が大きく変動する特徴があると分かるので、 $\alpha$ と $\beta$ の値を比較的大きくするべきである。 $U$ の分散が小さい信号であれば、普段から第1・第2成分の変動が小さいという特徴があると分かるので、 $\alpha$ と $\beta$ の値を比較的小さくするべきである。今回は、28日分のJPIXトラヒックを特異値分解して得られた $U$ から $\alpha$ と $\beta$ を求める場合を考える。ここで、実際の例を示しながら説明を行う。

2012年3月11日(日)~2012年4月7日(土)までの28日分のトラヒックを特異値分解して得られた $U$ から $\alpha$ と $\beta$ を求める。

図5.16に第1成分に対する $U$ を、図5.17に第2成分に対する $U$ をそれぞれ示す。

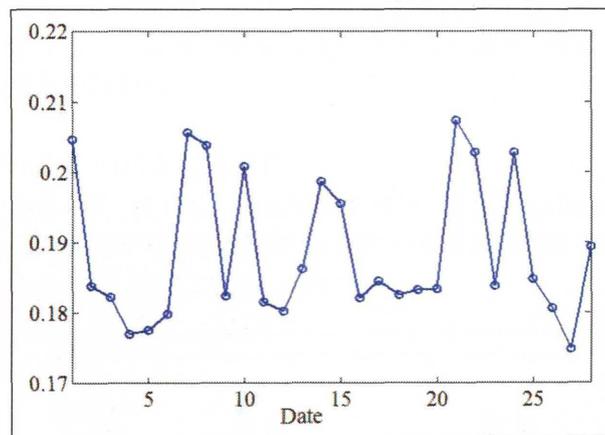


図 5.16: 第1成分に対する $U$

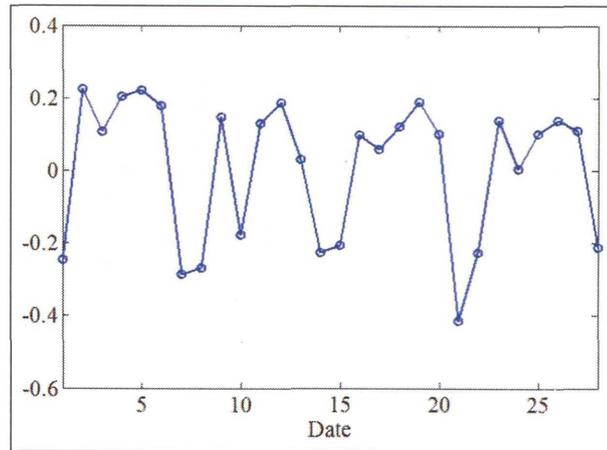


図 5.17: 第 2 成分に対する  $U$

図 5.16, 図 5.17 を見ると, 周期的に大きな変動が見られる事が分かる. これは第 3 章でも述べた通り, 平日における第 1・第 2 成分の大きさと, 休日における第 1・第 2 成分の大きさには大きな違いがあるためである. このような大きな変動が含まれたまま, 分散の計算を行ってしまうと, 必要以上に  $\alpha$  と  $\beta$  の値が大きくなってしまう. そこで, これから行う操作によって大きな変動部分を取り除く.

操作 1: 各点における  $U$  の差を計算する.

$U$  は 1 点~28 点までであるので, (1 点目-2 点目), (2 点目-3 点目), (3 点目-4 点目), ..., (27 点目-28 点目) をそれぞれ計算する. その結果 27 個の差を求める事ができる. 図 5.16, 図 5.17 において差を計算したものを, 図 5.18, 図 5.19 に示す.

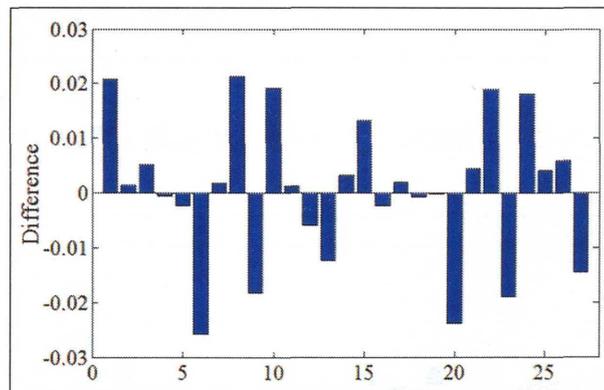


図 5.18: 第 1 成分に対する  $U$  の各点の差

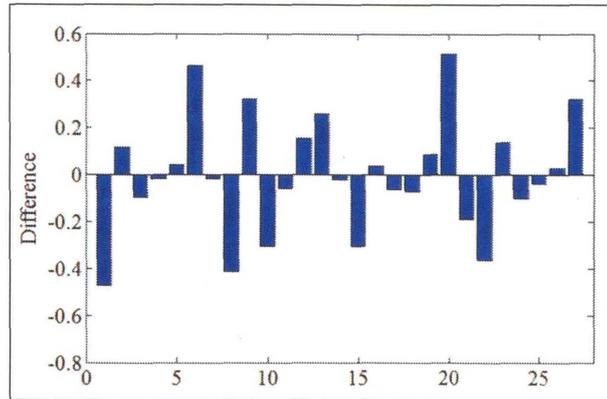


図 5.19: 第 2 成分に対する  $U$  の各点の差

図 5.18, 図 5.19 には, 大きな変動が含まれているので, 27 個の差のうち絶対値の小さい方から順に 14 個だけを取り出す. ここで出てきた 14 という数字は, 初めの  $U$  の点数 28 の半分から来ている. 絶対値の小さい方から 14 個を取り出したものを図 5.20, 図 5.21 に示す.

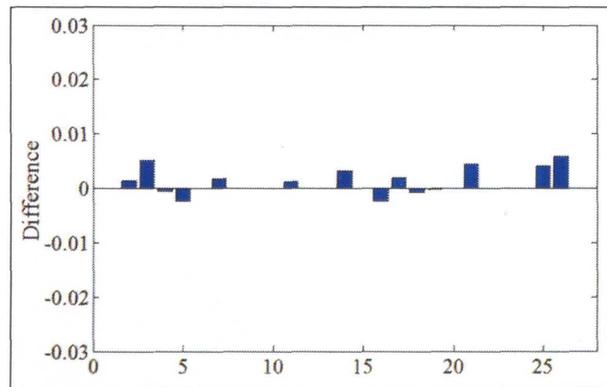


図 5.20: 第 1 成分の  $U$  の差のうち絶対値が小さい 14 個

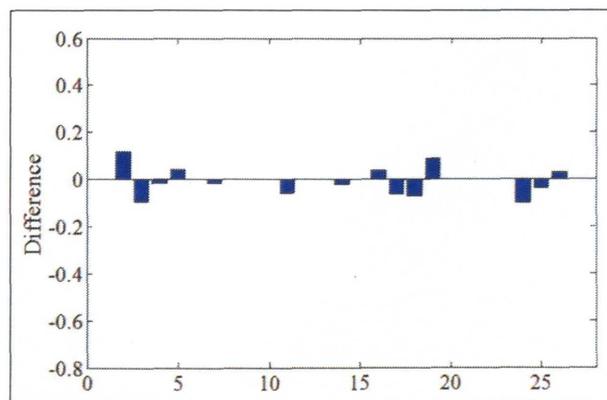


図 5.21: 第 2 成分の  $U$  の差のうち絶対値が小さい 14 個

図 5.20, 図 5.21 を見ると, 大きい変動を取り除けていることが分かる. この 14 個の値に対しての分散を  $\alpha, \beta$  とする. この時の第 1 成分の分散  $\alpha$ , 第 2 成分の分散  $\beta$  を実際に求めてみる.

計算に用いる分散の式は, 以下の式 (5.7) である.

$$\sigma^2(x) = \frac{1}{n} \{(x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + (x_3 - \bar{x})^2 + \cdots + (x_n - \bar{x})^2\} \quad (5.7)$$

式 (5.7) は, 配列  $x (x_1, x_2, x_3, \dots, x_n)$  の分散を求める場合の式を示している.  $\bar{x}$  は配列  $x$  における平均値を示している.

$$\alpha = 0.0000073$$

$$\beta = 0.0044$$

と求められた.  $\alpha$  の値が  $\beta$  に比べて遥かに小さい値になっているが, これは第 1 成分に対する  $U$  が変動しにくい為であり, 妥当な結果であると言える. このようにして求めた  $\alpha, \beta$  を用い, 第 5.1.2 章~第 5.1.4 章で説明した方法を用いて逐次更新を行う. 繰り返しになるが, JPIX のトラヒックは 5 分に 1 点得られるので, 5 分毎に予測トラヒックの更新を行う. 第 5.2 章から実際に予測トラヒックを逐次更新する実例を示す.

## 5.2 逐次更新の実例

第 5.2 章では, 逐次更新を行った例を実際に示す.

### 5.2.1 逐次更新手法

予測トラヒックの逐次更新に用いた手法は, 以下の通りである.

1. 予測日の前日までに, 予測トラヒックを作成しておく.
2. 予測日当日の 0 時になり, 実際のトラヒックが 5 分に 1 点得られる.
3. 0 時~6 時までは何も行わない.
4. 6 時になり, 6 時間分 (72 点分) のトラヒックが得られた時逐次更新を開始.
5. 5 分毎に 1 点, 実際のトラヒックを得る.
6. 現在の予測トラヒックを基準として, ランダムサーチを行い, 予測トラヒックと実際のトラヒックを合わせ込む.
7. 手順 6 において合わせこんだ予測トラヒックを, 新たな予測トラヒックとして採用.
8. 6 時~24 時の間, 手順 5~7 を繰り返す.

手順 3 において, 0 時~6 時まで何も行わないのは, 逐次更新を行うのに十分なトラヒックデータを得るためである. トラヒックデータが少ない状態において, 予測トラヒックと実際のトラヒックの合わせ込みを行うと, 予測トラヒックが想定外の特徴を持つ事がある. 予測トラヒックが想定外の特徴を持つと, 1 日全体として見た時に, 予測トラヒックと実際のトラヒックの相対誤差が大きくなってしまふ. その例を, 図 5.22 に示す.

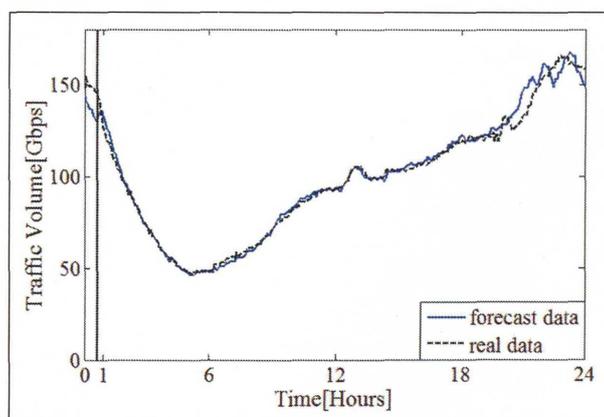


図 5.22: 1 日の初めがに誤差が出ている場合 (合わせ込み前)

図 5.22 において, 実線が予測トラヒック, 破線が実際に得られたトラヒックである. この図は, 予測日当日になり実際のトラヒックデータが 40 分得られた所を示している. 図 5.22 において, 縦線が 0 時 40 分を示しているので, 縦線の左側までのトラヒックデータが得られて

おり、縦線の右側のトラフィックデータは未知である。縦線の左側を見ると、大きな誤差が出ている事が分かるが、1日全体として見た時には比較的良好な予測が出来ている。このような場合に、縦線の左側だけを基準にして予測トラフィックを合わせこんだ場合を、図 5.23 に示す。

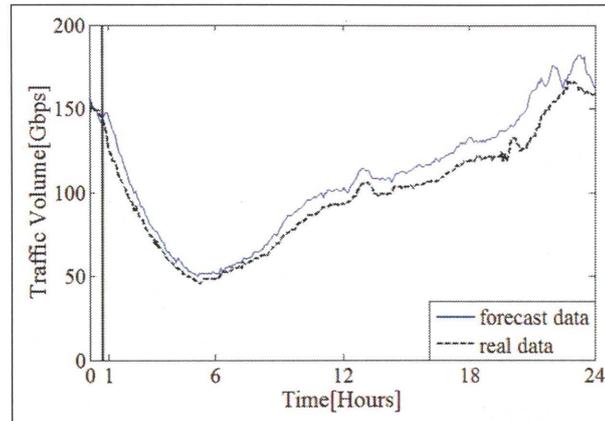


図 5.23: 1日の初めがに誤差が出ている場合 (合わせ込み後)

図 5.23 を見ると、縦線の左側の誤差は小さくなっているが、1日全体として見た時に予測トラフィックと実際のトラフィックに大きな誤差が生じていることが分かる。このように、得られたトラフィックデータが少ない状態で合わせ込みを行うことは、誤差を大きくしてしまう可能性が高い。その為、一定量のトラフィックデータが得られるまで待つのであり、本研究においては6時間(72点分)のデータが得られるまで待つのである。

## 5.2.2 実際のデータにおける逐次更新の実例

2012年3月11日(日)~2012年4月7日(土)までの28日分のトラフィックを用いて、2012年4月8日(日)の予測トラフィックを作成する。2012年4月8日の予測日当日になり、0時~6時までのトラフィックが得られた時点で予測トラフィックの逐次更新を開始する。図5.24に、更新する前の予測トラフィックと実際のトラフィックを重ねて表示したものを示す。

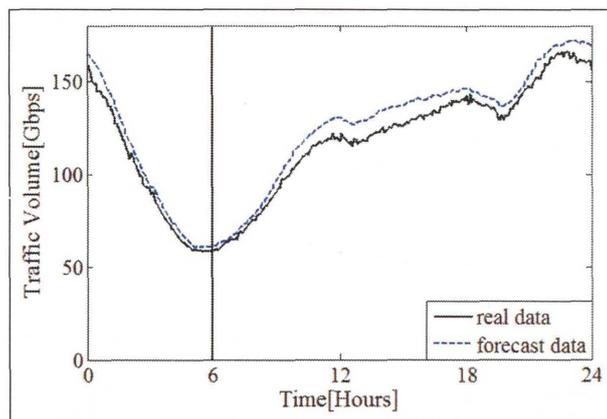


図 5.24: 更新を開始する前

図5.24は更新を開始する前の画像である。実線が実際のトラフィックを、破線が予測トラフィックを表している。0時~6時までのトラフィックデータが得られるまでは、予測トラフィックの逐次更新を行わない。縦線は6時を示しているので、縦線までのトラフィックデータが得られた時に更新を開始する。次の図5.25~図5.28に、逐次更新を開始した後の、各時刻における予測トラフィックと実際のトラフィックを重ねて表示したものを示す。

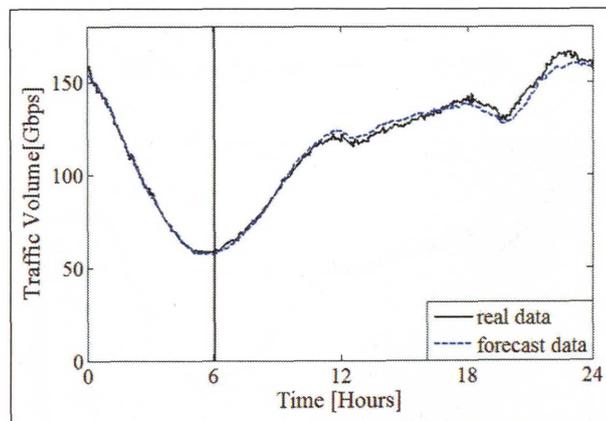


図 5.25: 6時時点における更新

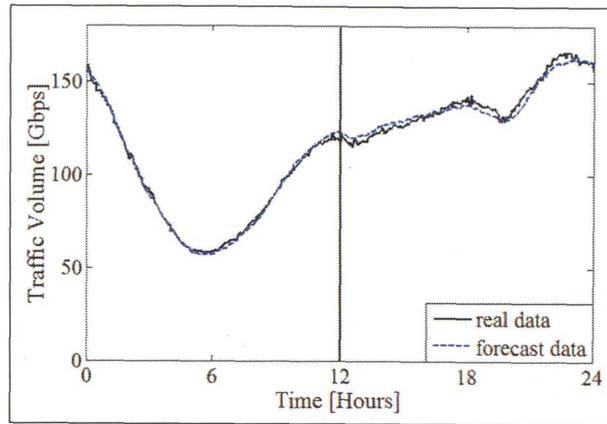


図 5.26: 12 時時点における更新

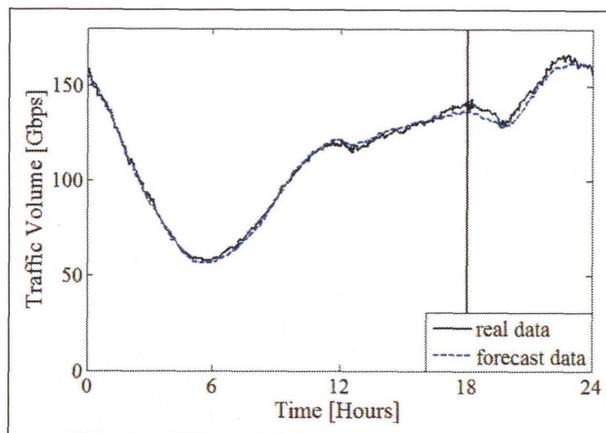


図 5.27: 18 時時点における更新

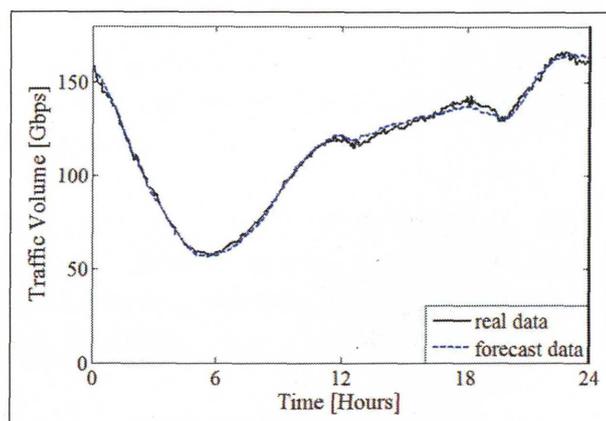


図 5.28: 24 時時点における更新

図 5.25 を見ると, 更新を開始した直後の 6 時時点で, かなり誤差が小さくなっていること

が見て取れる。縦線は6時を表しており、縦線の左側が既知のデータ、縦線の右側が未知のデータである。既知のデータに予測トラヒックを上手く合わせこんだので、当然縦線の左側は上手く合っているが、未知のデータである縦線の右側についても上手く合っている事が見て取れる。これは、既知の6時間分のデータから特徴を読み取り、予測トラヒックに実際のトラヒックの特徴を上手く取り込めた為である。

また、図 5.26～図 5.28 を見ると、得られたトラヒックデータが増えれば増えるほど僅かずつではあるが、誤差が小さくなっていることが分かる。目視では、相対誤差がどの程度下がっているのかを読み取るのが難しいので、各時間帯における予測トラヒックと実際のトラヒックの相対誤差を表したものを図 5.29 に示す。ここで、相対誤差は第 4 章の式 (4.4) で示したものと同一ものを使って求めている。

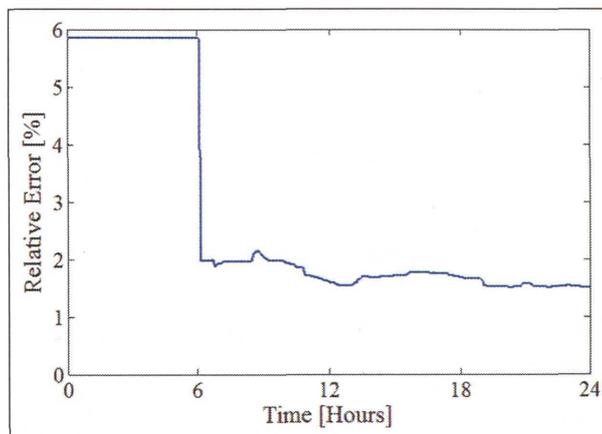


図 5.29: 各時刻における相対誤差

図 5.29 の相対誤差の読み取り方について説明する。縦軸は相対誤差を表しており、横軸は時刻を表している。予測トラヒックの逐次更新を行うと、予測トラヒックが5分毎に変化し、実際のトラヒックとの相対誤差も時刻によって変わってくる。この図 5.29 は、5分毎に変化していく『予測トラヒックと実際のトラヒックの相対誤差』を各時刻において表したものである。0時～6時において全く相対誤差が変化していないが、これは予測トラヒックを逐次更新していない為であり、6時から逐次更新を開始するので、6時以降の相対誤差には変化が見られる。以上がこの図の読み取り方である。

なお、この相対誤差は1日分のトラヒック全体で求めている。既知の部分だけを用いて求めたものでも、未知の部分だけを用いて求めたものでもない事に注意して欲しい。図 5.29 を見ると、逐次更新を開始した直後に大幅な相対誤差現象が見られる。このように、逐次更新を行う事で予測トラヒックと実際のトラヒックの相対誤差を小さくできるので、逐次更新を行うのである。

### 5.2.3 実際のデータにおける逐次更新の実例 2

逐次更新の例が1つでは少ないので、もう1例挙げる。2012年3月12日(月)～2012年4月8日(日)までの28日分のトラフィックを用いて、2012年4月9日(月)の予測トラフィックを作成する。更新を開始する前の予測トラフィックと実際のトラフィックを図5.30に示す。

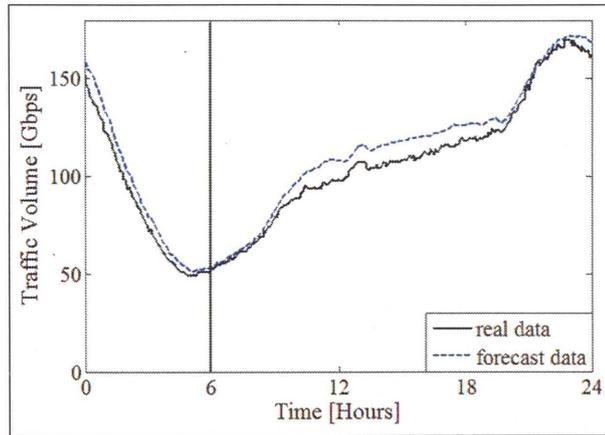


図 5.30: 更新を開始する前

0～6時までのトラフィックデータが得られ、更新を開始した後の各時刻における予測トラフィックと実際のトラフィックを重ねて表示したものを図5.31～図5.34に示す。

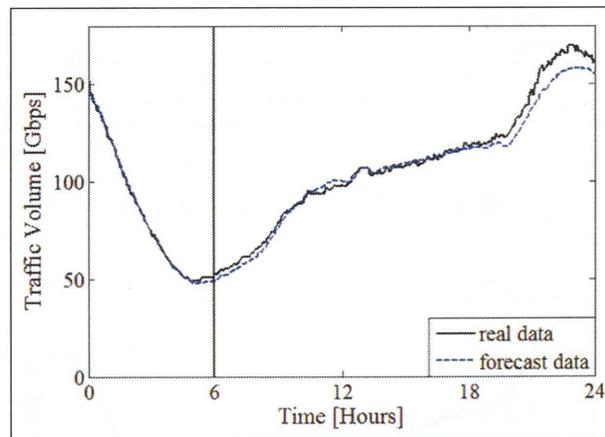


図 5.31: 6時時点における更新

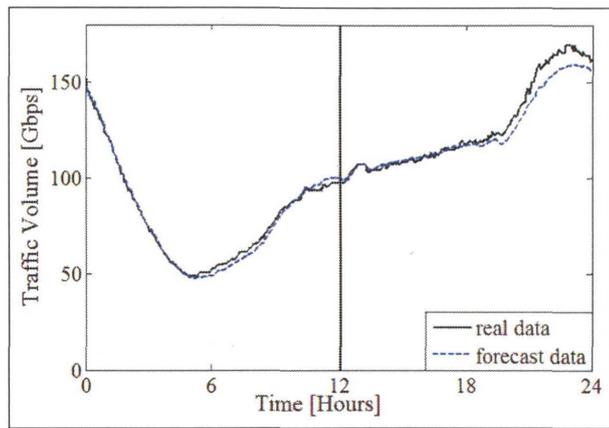


図 5.32: 12 時時点における更新

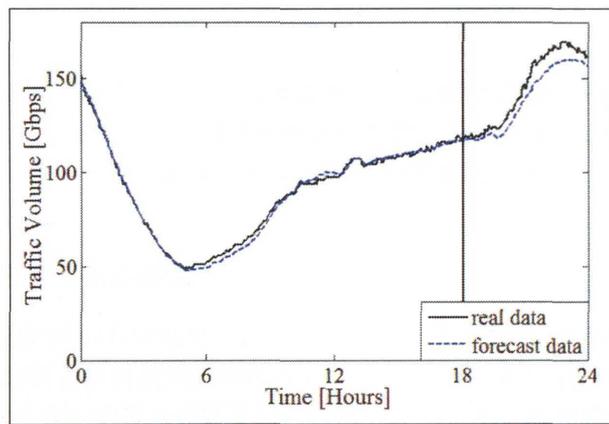


図 5.33: 18 時時点における更新

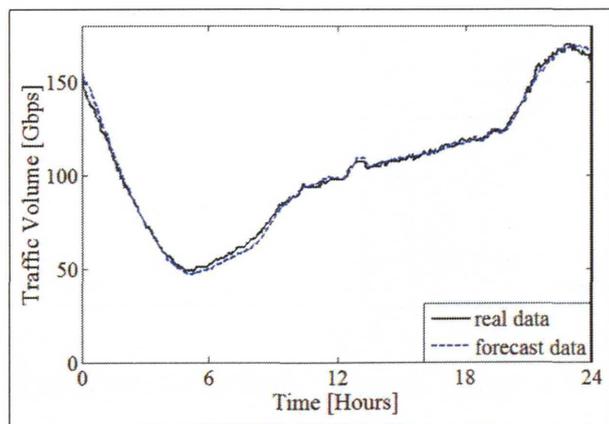


図 5.34: 24 時時点における更新

この例における、各時刻の予測トラフィックと実測トラフィックの相対誤差を示したものを図

5.35 に示す.

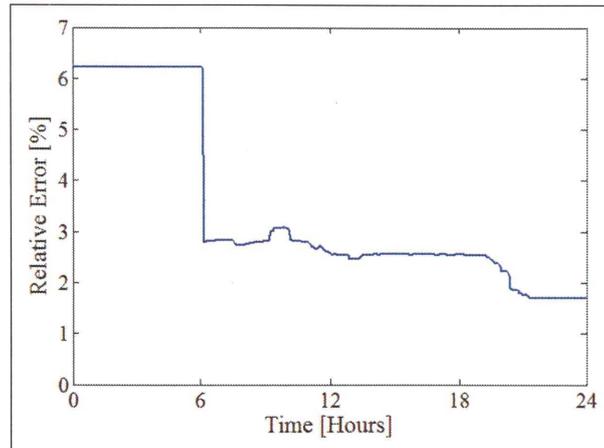


図 5.35: 各時刻における相対誤差

図 5.35 を見ると, この例においても逐次更新によって相対誤差を大幅に小さくする事に成功していると分かる. 2 例だけでは, 逐次更新法の有効性を検証することができないので, この後の章において, 多数のデータに逐次更新を行った結果を示す.

#### 5.2.4 逐次更新法の有効性検証

この章では, 逐次更新法の有効性について検証する. 逐次更新法を 30 日分の予測トラヒックに対して適用し, 時刻と相対誤差の関係を求めた. 2012 年 4 月 10 日 (火)~2012 年 5 月 9 日 (水) のトラヒックに対して逐次更新を行った時の時刻と相対誤差の関係を求めたが, ここでは代表例として 2012 年 4 月 10 日 (火)~2012 年 4 月 15 日 (日) のグラフを図 5.36~図 5.41 に示す. 2012 年 4 月 16 日 (月)~2012 年 5 月 9 日 (水) のグラフについては, 付録 D に示す.

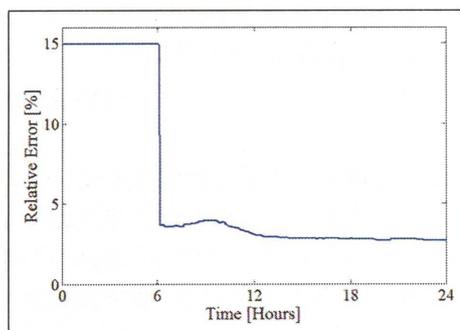


図 5.36: 2012 年 4 月 10 日 (火) の相対誤差

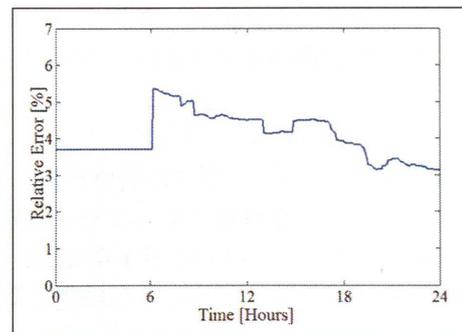


図 5.37: 2012 年 4 月 11 日 (水) の相対誤差

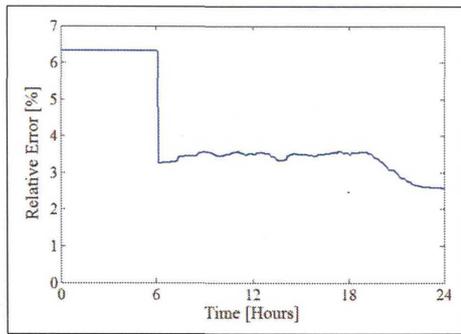


図 5.38: 2012 年 4 月 12 日 (木) の相対誤差

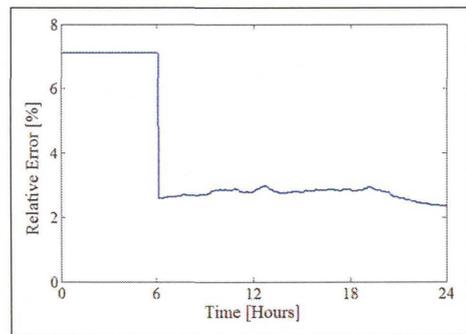


図 5.39: 2012 年 4 月 13 日 (金) の相対誤差

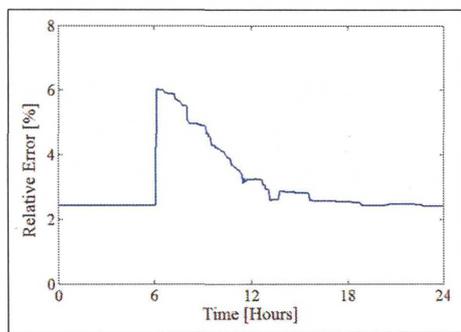


図 5.40: 2012 年 4 月 14 日 (土) の相対誤差

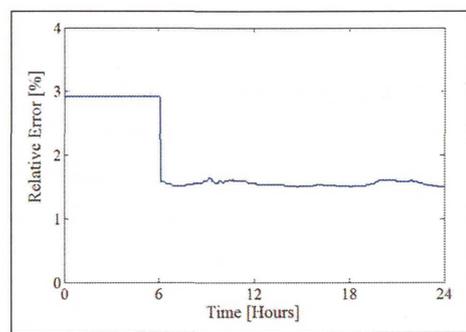


図 5.41: 2012 年 4 月 15 日 (日) の相対誤差

図 5.35～図 5.41 を見ると、相対誤差と時刻の関係には大きく分けて 2 つのパターンがあると分かる。1 つめのパターンは、更新開始と共に急激に相対誤差が下がるものである。例えば、図 5.36、図 5.38 などである。2 つめのパターンは、更新開始と同時に相対誤差があがり、その後緩やかに減少していくものである。例えば、図 5.37、図 5.40 などである。逐次更新によって相対誤差が大きくなってしまふパターン 2 は好ましくないと言える。30 日分の逐次更新のうち、パターン 1 が 23 日分、パターン 2 が 7 日分あり、パターン 2 の数は決して少ないとは言えない。

パターン 2 の原因として、予測トラヒックの逐次更新を行った際、1 日の前半部分のトラヒックは上手く合うようになるが、1 日の後半部分のトラヒックと合わなくなってしまう事が挙げられる。このことを、パターン 2 が見られた 2012 年 4 月 14 日 (土) の逐次更新の例を使って説明する。逐次更新する前の予測トラヒックと実際のトラヒックを重ねて表示したものを図 5.42 に示す。

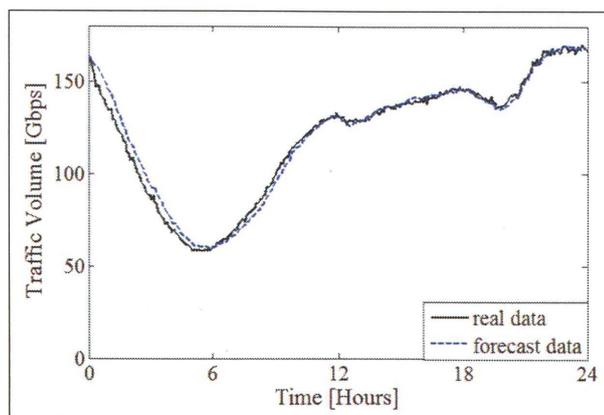


図 5.42: 2012 年 4 月 14 日 (土) : 更新前

図 5.42 を見ると, 1 日の前半の部分に多少の誤差が見られるが, 比較的良好な予測が出来ている事が読み取れる. 0 時~6 時までのトラフィックデータが得られたとして, 逐次更新を開始した直後の予測トラフィックと実際のトラフィックを図 5.43 に示す.

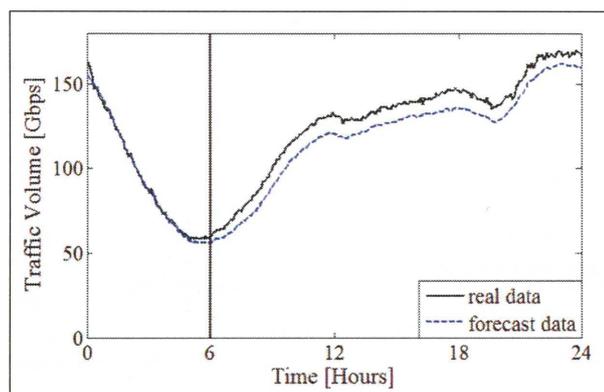


図 5.43: 2012 年 4 月 14 日 (土) : 更新開始直後

図 5.43 を見ると, 1 日の前半の部分 (既知の部分) に予測トラフィックを合わせこんだ為に, 1 日の後半の部分の予測に大きな誤差が生じている事が読み取れる. このように, 1 日の前半部分で予測トラフィックを合わせ込むと, 1 日の後半部分に大きな誤差が生じる場合が多々見られる. この対策として, 未知の部分 (1 日の後半部分) においては, 逐次更新の影響をあまり受けなくするという方法がある. つまり, 既知の部分 (1 日の前半部分) では逐次更新による影響を大いに受け, 未知の部分 (1 日の後半部分) においては逐次更新開始前のトラフィックのままにしておくという事である. この方法については, 次の章で説明する事とする.

## 第6章 忘却型逐次更新

第5章で述べた通り, 単純に既知部分のトラフィックに予測トラフィックを合わせこんでしまうと, 1日の後半の予測トラフィックが大きく変動し, 誤差が大きくなってしまう可能性がある. そこで, 既知部分については予測トラフィックを実際のトラフィックに合わせ込み, 未知部分については予測トラフィックを1日の初めの予測トラフィックからあまり変動させないようにするという方法を取る. このことを, 『忘却型逐次更新』と呼ぶこととする. このことを分かり易く表現した図を, 図6.1に示す.

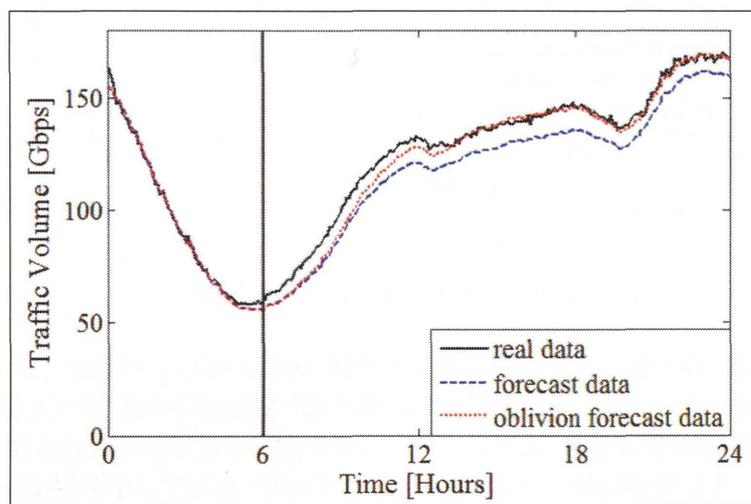


図 6.1: 逐次更新と忘却型逐次更新の比較

図6.1は, 6時における実際のトラフィック, 逐次更新トラフィック, 忘却型逐次更新トラフィックの3つを表している. 実線が実際のトラフィック, 粗い点線が逐次更新トラフィック, 細かい点線が忘却型逐次更新トラフィックを表している. 実際のトラフィックにおいて6時の時点で既知なのは縦線の左側, 未知なのは縦線の右側である. 図6.1を見ると, 既知のトラフィックに予測トラフィックを上手く合わせこんだ結果, 逐次更新トラフィックは未知の部分に大きな誤差が出ている事が分かる. それに比べ, 忘却型逐次更新トラフィックは未知の部分において更新の影響を忘却するようになっている為, 未知の部分の誤差が小さくなっている事が分かる. この第6章では, 忘却型逐次更新の方法を述べ, 実際の実行結果を示す事により, この方法の有効性を示す.

## 6.1 基本方針

忘却型更新を簡単に説明すると、『逐次更新を行った予測トラフィック』と『1日の初めの予測トラフィック』のブレンドである。予測トラフィックの逐次更新を行い、既知の部分においては逐次更新のトラフィックをそのまま用い、未知の部分においては逐次更新のトラフィックと1日の初めの予測トラフィックを混ぜ合わせたものを用いる。この事を図6.2に示す。

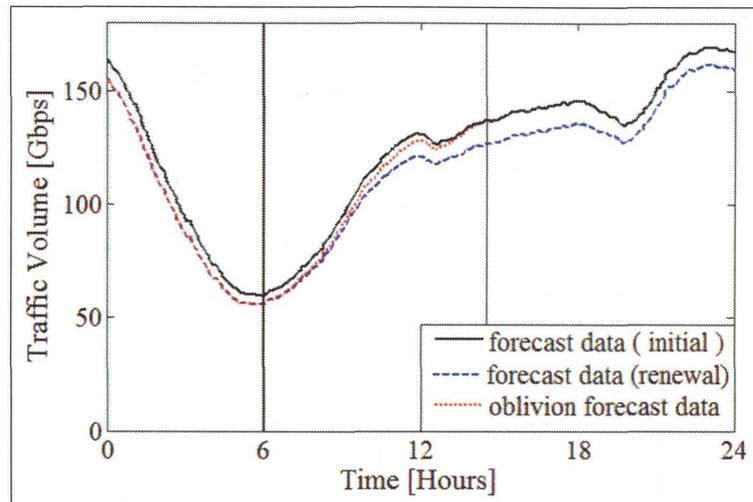


図 6.2: 忘却型逐次更新の説明 1

図6.2において、実線が『1日の初めにおける予測トラフィック』を、粗い破線が『逐次更新の予測トラフィック』を、細かい破線が『忘却型逐次更新の予測トラフィック』を表している。図6.2は、実際に得られたトラフィックの波形を示していないが、6時まで実際のトラフィックが得られた時の逐次更新を示している。図6.2の0時～6時までの波形を見ると、『逐次更新』と『忘却型逐次更新』の波形が一致している事が分かる。これは、『忘却型逐次更新』は既知の部分については『逐次更新』のトラフィックをそのまま用いる為である。図6.2の6時～14時頃までの波形を見ると、『忘却型逐次更新の波形』が『逐次更新の波形』から『1日の初めの予測の波形』へとだんだんと近づいている事が分かる。これは、『忘却型逐次更新』は『逐次更新』による影響を8時間20分(トラフィック100点分)の時間をかけてだんだんと忘却していくように設計した為である。図6.2の14時頃～24時までの波形を見ると、『忘却型逐次更新の波形』が『1日の初めの予測の波形』と一致している事が分かる。これは、『忘却型逐次更新』が『逐次更新』による影響を完全に忘却した為である。この事を分かり易く示したものを図6.3に示す。

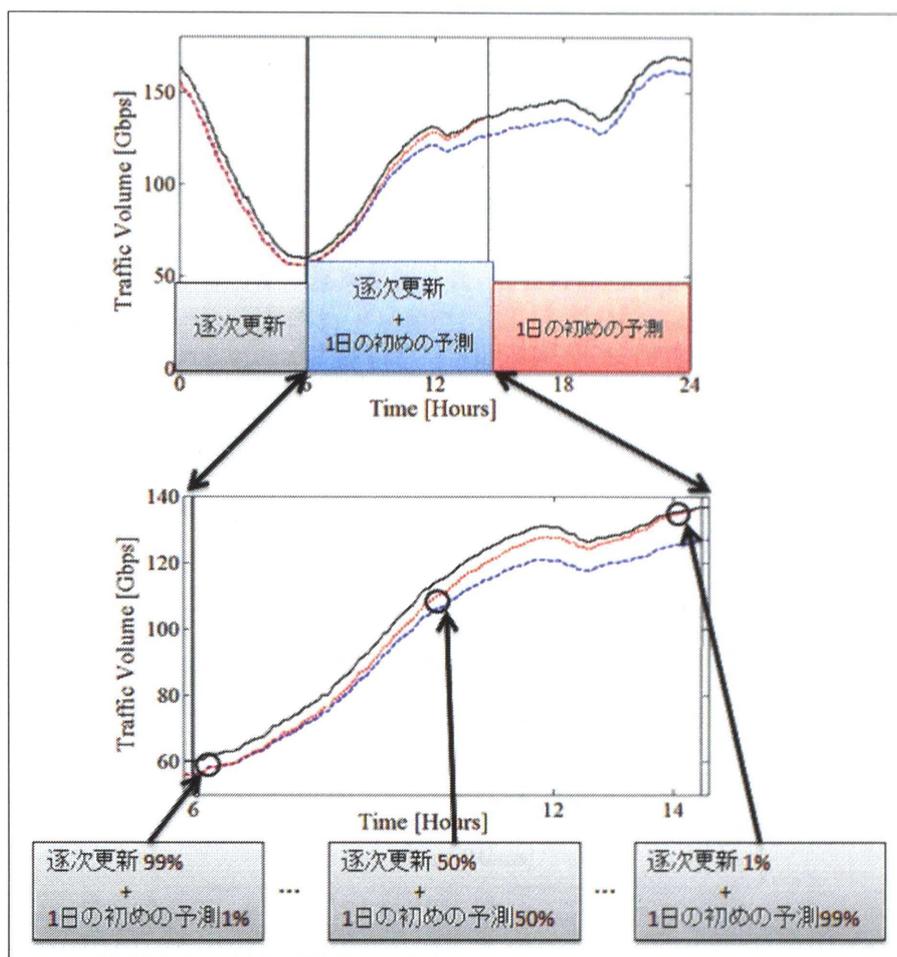


図 6.3: 忘却型逐次更新の説明 2

図 6.3 のように、忘却型逐次更新においては、既知の部分に逐次更新の波形を、未知の 8 時間 20 分に逐次更新と 1 日の初めの予測の波形を混ぜたものを、それ以降に 1 日の初めの予測の波形を用いる。また、図 6.2 で示したように波形を混ぜ合わせる際には、それぞれの比率を少しずつ変えていく。波形を混ぜ合わせる部分は 100 点分あるので、逐次更新の割合を 1% ずつ減らし、1 日の初めの予測の割合を 1% ずつ増やして行く。式で表すと以下のようになる。

1 点目 = 逐次更新 99% + 1 日の初めの予測 1%

2 点目 = 逐次更新 98% + 1 日の初めの予測 2%

...

100 点目 = 逐次更新 1% + 1 日の初めの予測 99%

このようにして、逐次更新の影響を 1% ずつ減らし、逐次更新の影響を忘却するようになっている。ここで、既知部分は実際のトラフィックが得られるほどに増えていくが、それに伴って、忘却部分も後ろに下がっていく事になる。つまり、6 時まで実際のトラフィックが得られている時、忘却を行う部分は 6 時～14 時 20 分であるが、7 時まで実際のトラフィックが得られたとすると、忘却を行う部分は 7 時～15 時 20 分となるようにどんどん後ろに下がっていく。JPIX

のトラヒックデータは5分に1点得られるので、5分毎に忘却部分は1点後ろに下がっていく。以上が忘却型逐次更新の基本的な仕様である。この次の章では、どのようにして『逐次更新』と『1日の初めの予測』を混ぜ合わせているかを説明する。

## 6.2 忘却度

この章では、どのようにして『逐次更新』と『1日の初めの予測』を混ぜ合わせているかを説明する。我々が用いた方法では、『忘却度』というものを定義して、逐次更新の影響を忘却し、『逆忘却度』というものを定義して、1日の初めの予測の影響を大きくしていく。

忘却度とは、0~1の数字を用いて表され、例えば以下の図6.4のようになる。

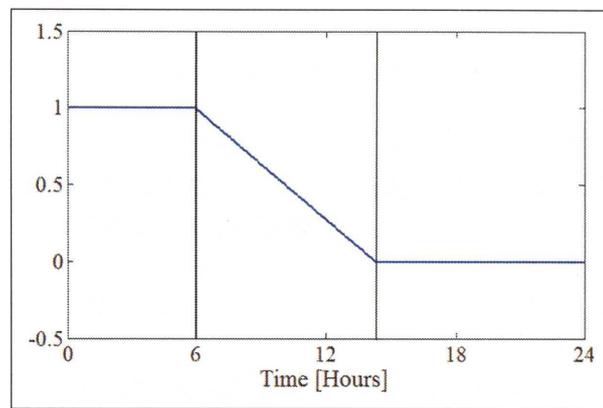


図 6.4: 6時における忘却度

図 6.4 は忘却度を表しており、横軸は時刻である。忘却度は時刻とともに変化し、これは6時における忘却度である。忘却度は『忘却型逐次更新』における『逐次更新波形』の影響度を表しており、1であれば100%の影響を及ぼし、0.5であれば50%の影響を及ぼし、0であれば0%の影響を及ぼすようになっている。例えば、0時~6時において忘却度は1になっており、この部分は100%『逐次更新波形』であることが分かる。また、14時20分以降は全て0になっており、この部分には『逐次更新波形』が全く含まれていない事が分かる。

6時~14時20分において、忘却度が徐々に下がって行くようになっているが、この部分が、『逐次更新』と『1日の初めの予測』を混ぜ合わせる部分である。100点分で一次関数的に忘却度が下がるようになっているので、1点につき0.01ずつ忘却度が減っていることになる。つまり、1.00, 0.99, 0.98, 0.97, ..., 0.01, 0.00と値が下がっていくようになっている。この部分では、『逐次更新波形』による影響が少しずつ小さくなっている事が分かる。次に、『逆忘却度』について説明する。

逆忘却度は、1 から『忘却度』の値を引いて求められる。逆忘却度の例を、図 6.5 に示す。

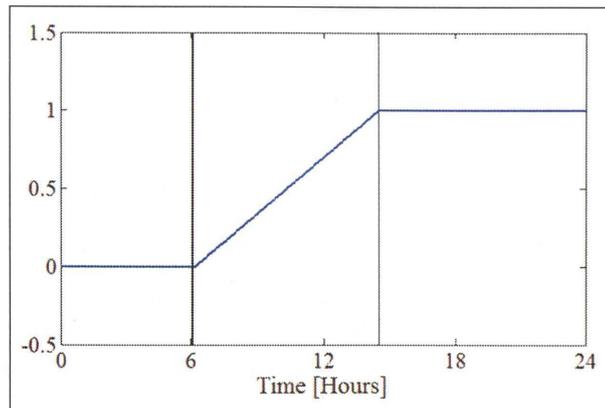


図 6.5: 6時における逆忘却度

図 6.5 は逆忘却度を表しており、横軸は時刻である。忘却度は『忘却型逐次更新』における『1日の初めの予測』の影響度を表しており、1であれば100%の影響を及ぼし、0.5であれば50%の影響を及ぼし、0であれば0%の影響を及ぼすようになっている。0時～6時において逆忘却度は0になっており、この部分には『1日の初めの予測』が全く含まれていない事が分かる。また、14時20分以降は全て1になっており、この部分は100%『1日の初めの予測』である事が分かる。

6時～14時20分において、逆忘却度が徐々に上がっていくようになっているが、この部分が、『逐次更新』と『1日の初めの予測』を混ぜ合わせる部分である。100点分で一次関数的に逆忘却度が上がるようになっているので、1点につき0.01づつ忘却度が上がっていることになる。つまり、0.00, 0.01, 0.02, 0.03, …, 0.99, 1.00と値が上がっていくようになっている。この部分では、『1日の初めの予測』による影響が少しずつ大きくなっていく事が分かる。以上が逆忘却度の説明である。

前述の通り、忘却度と逆忘却度は時間経過によって変化する。逆忘却度については、忘却度から算出する事ができるので、忘却度の時間経過による変化について説明する。忘却度は、実際に得られたトラフィックデータが増えれば増えるほど右にスライドして行き、24時になると全てが1になる。このことを図で表すと、図 6.6 のようになる。

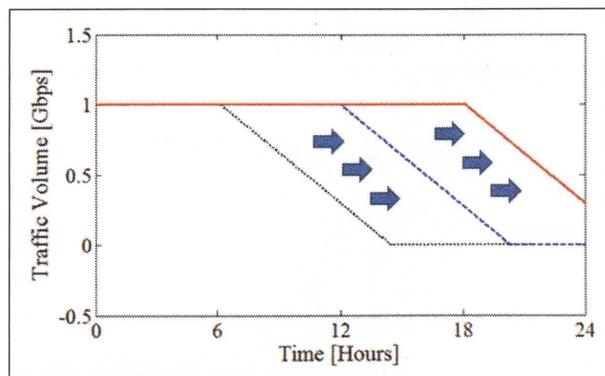


図 6.6: 忘却度の時間的变化

図 6.6 において、矢印は時間経過を表している。このようにして、時間が経過するごとに忘却度は変化していく。

この忘却度と逆忘却度を用いて、逐次更新と 1 日の初めの予測を混ぜ合わせるのである。混ぜ合わせる方法については、次の章で説明を行う。

### 6.3 忘却型逐次更新の実現

先ほど説明した『忘却度』と『逆忘却度』を用いて、忘却型逐次更新を実現する。忘却型逐次更新は非常に単純な方法で実現されており、その方法とは『忘却度と逐次更新を掛け合わせたもの』と『逆忘却度と 1 日の初めの予測を掛け合わせたもの』を足し合わせるだけである。このことを式にすると、次の式 6.1 である。

$$\text{忘却型逐次更新} = (\text{忘却度} \times \text{逐次更新}) + (\text{逆忘却度} \times \text{1日の初めの予測}) \quad (6.1)$$

このことを図で表すと、図 6.6～図 6.8 になる。

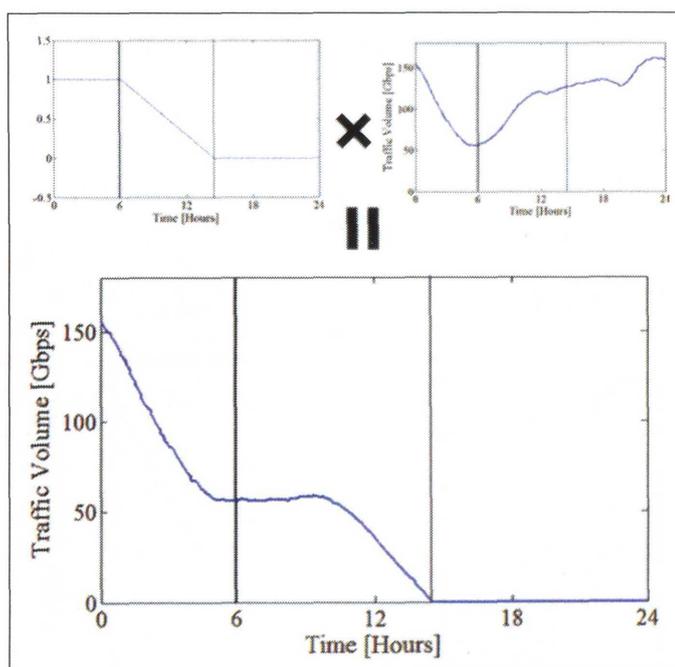


図 6.7: 忘却度 × 逐次更新

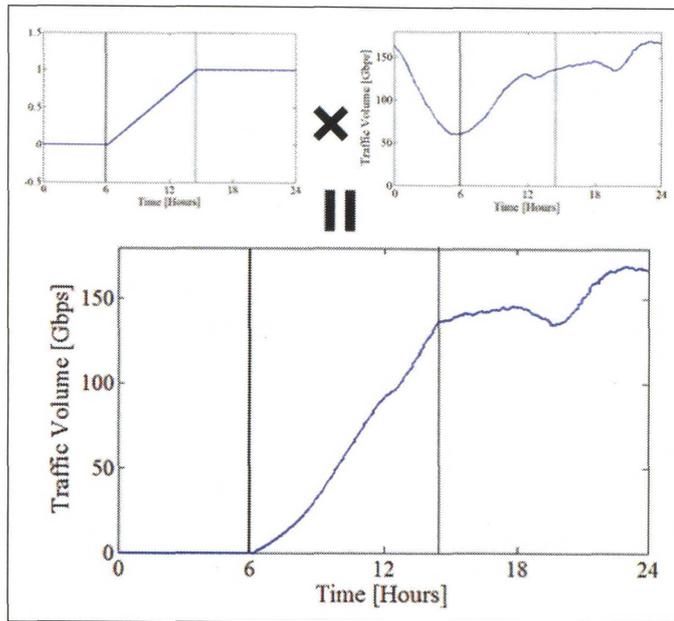


図 6.8: 逆忘却度 × 1 日の初めの予測

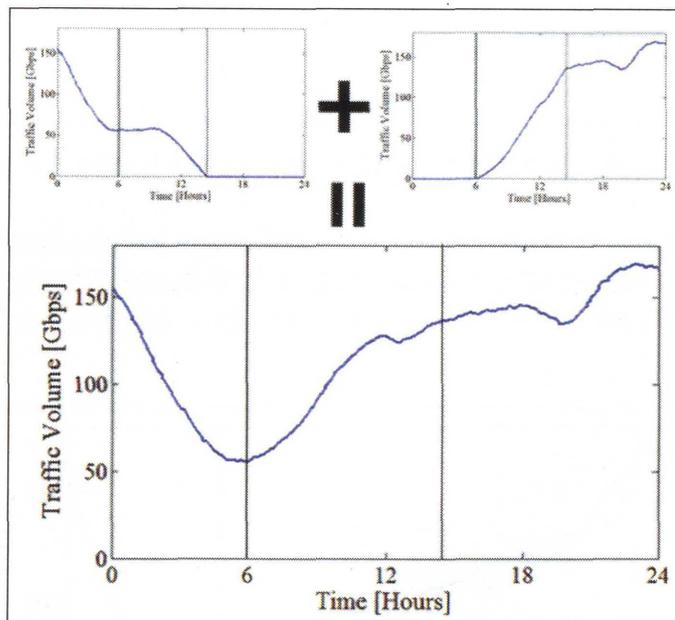


図 6.9: 忘却型逐次更新の実現

図 6.6～図 6.8 のようにして、忘却型逐次更新を実現している。忘却型逐次更新は、逐次更新波形を用いるため、逐次更新を行った後に行う操作になる。また、逐次更新と同様にあらたなトラフィックデータが得られる度に行い、JPIX のトラフィックデータは 5 分に 1 点得られるので、5 分毎にこのような操作を行う。次の章において、忘却型逐次更新を行った実際の例を示し、逐次更新の波形と比較する。

## 6.4 忘却型逐次更新の実例

この章では、忘却型逐次更新を実際に行った例を示す。忘却型逐次更新は、逐次更新において既知部分で合わせ込みを行った場合に、未知部分に大きな誤差が発生するのを防止する為の手法である。その為、ここではそのような場合をあえて選んで実例を示す。第5.2.4章を見ると、2012年4月1日(水)、2012年4月14日(土)においてそのような特徴が見られる。そこで、この2日分における『実際のトラフィック』と『忘却型逐次更新』と『逐次更新』の波形を重ねて表示したものを示す。

2012年4月1日(水)における逐次更新と忘却型逐次更新の波形を重ねて表示したものを次の図6.10～図6.14に示す。

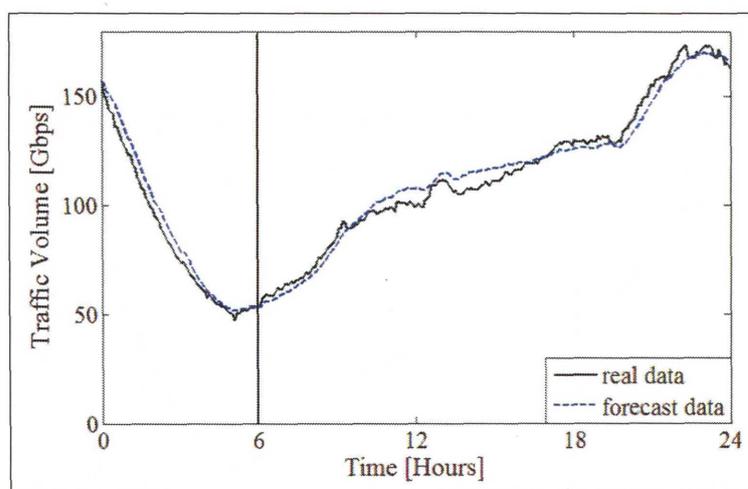


図 6.10: 2012年4月1日(水)における予測トラフィック(更新前)

図6.10は、逐次更新を開始する前の予測トラフィックと実際のトラフィックを重ねて表示したものである。逐次更新を開始する前では、逐次更新も忘却型逐次更新も行われないので、予測トラフィックは1つだけである。0時～6時までのトラフィックデータが得られると、更新を開始する。

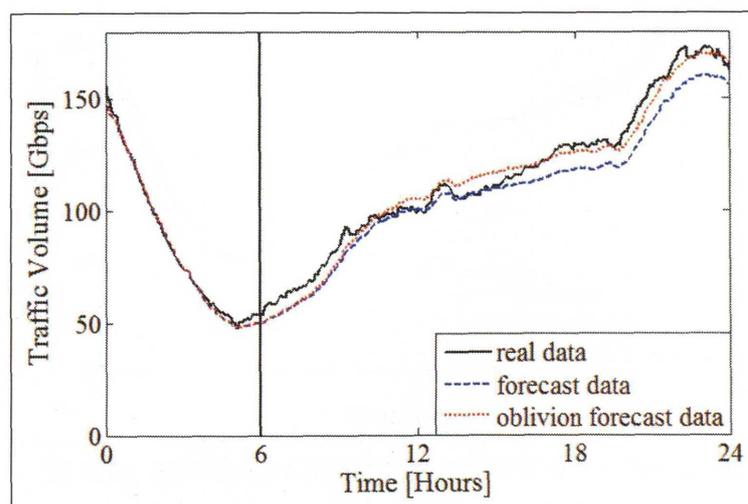


図 6.11: 6時における逐次更新と忘却型逐次更新の比較

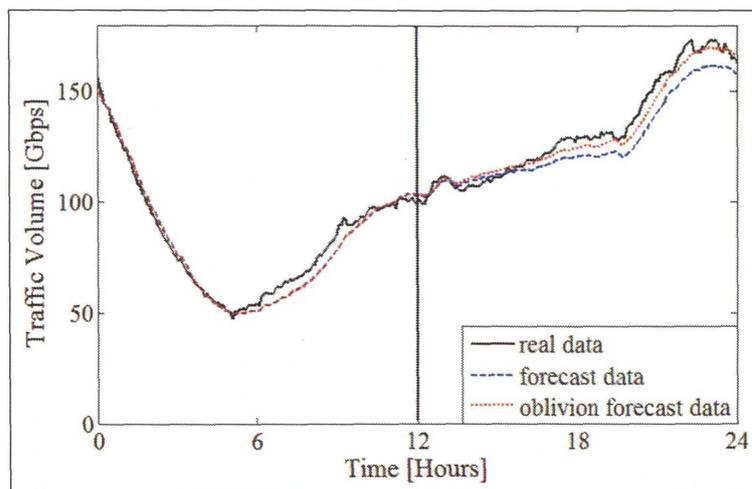


図 6.12: 12 時における逐次更新と忘却型逐次更新の比較

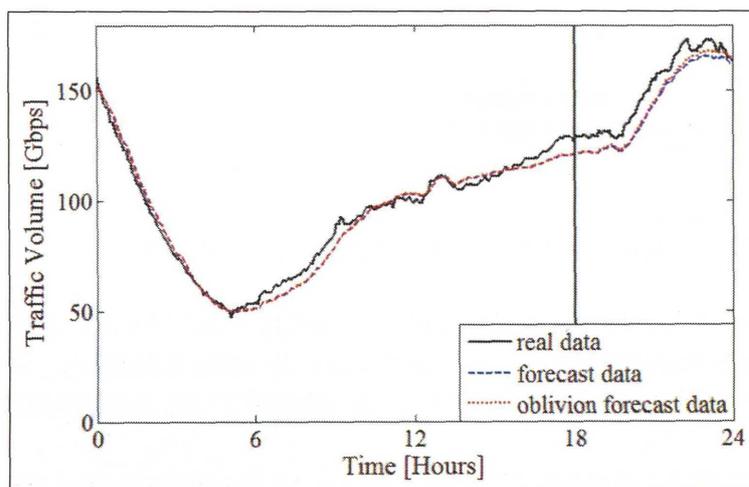


図 6.13: 18 時における逐次更新と忘却型逐次更新の比較

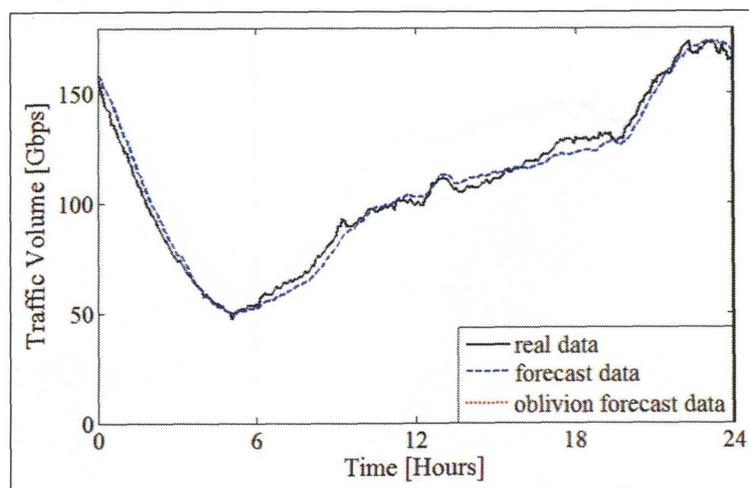


図 6.14: 24 時における逐次更新と忘却型逐次更新の比較

忘却型逐次更新はその性質上、24 時における波形が逐次更新と一致する。図 6.14 を見ると、忘却型逐次更新の波形が見えないが、それは逐次更新の波形と重なっている為である。

図 6.11 を見ると、0 時～6 時の既知部分に上手く合わせこんだ結果、逐次更新では未知部分に大きな誤差が生じていることが分かる。それに比べて忘却型逐次更新では未知部分にそれほど大きな誤差が出ていない事が分かる。図 6.12 においても同様の事が読み取れる。

忘却型逐次更新と逐次更新の相対誤差を比較する為に、相対誤差と時刻の関係を図 6.15 に示す。

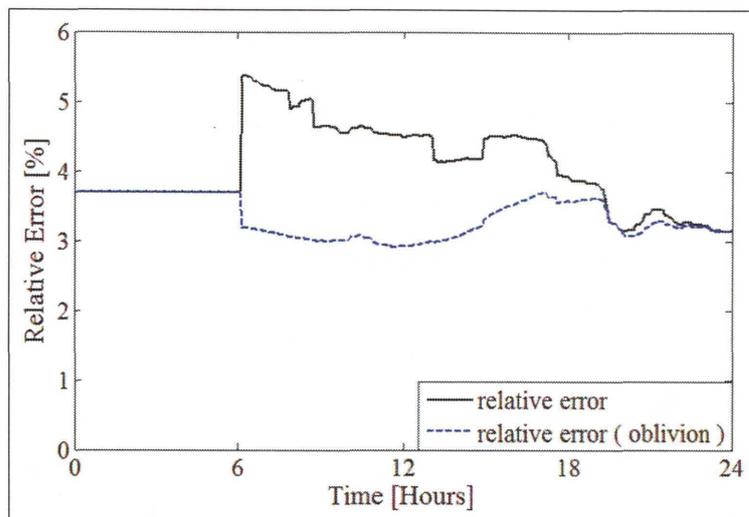


図 6.15: 逐次更新と忘却型逐次更新における相対誤差の比較

図 6.15 の実線は逐次更新の相対誤差を、破線は忘却型逐次更新の相対誤差を表している。図 6.15 を見ると、忘却型逐次更新の方は相対誤差の増加を防止する事が出来ており、6 時の忘却型逐次更新開始時点において相対誤差を減少させる事に成功していると分かる。

もう一つの例として、2012 年 4 月 14 日 (土) における逐次更新と忘却型逐次更新の波形を重ね合わせて表示したものを次の図 6.16～図 6.20 に示す。

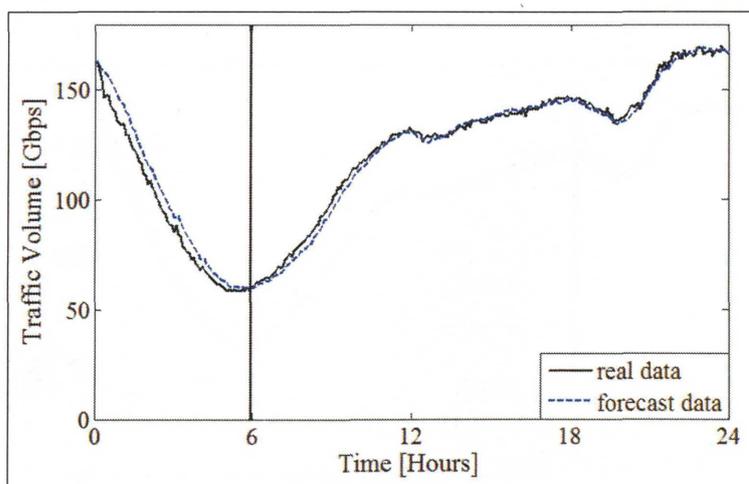


図 6.16: 2012 年 4 月 14 日 (土) における予測トラフィック (更新前)

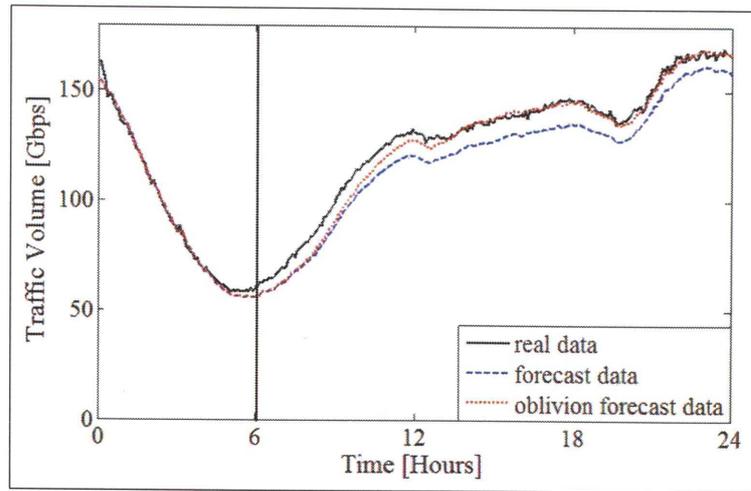


図 6.17: 6 時における逐次更新と忘却型逐次更新の比較

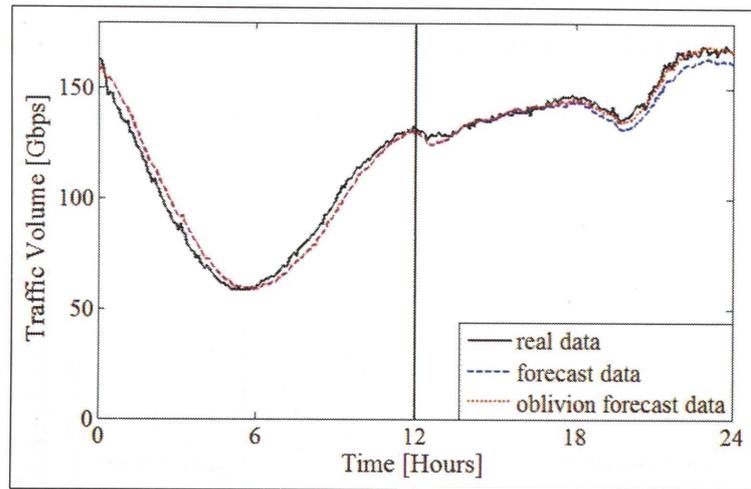


図 6.18: 12 時における逐次更新と忘却型逐次更新の比較

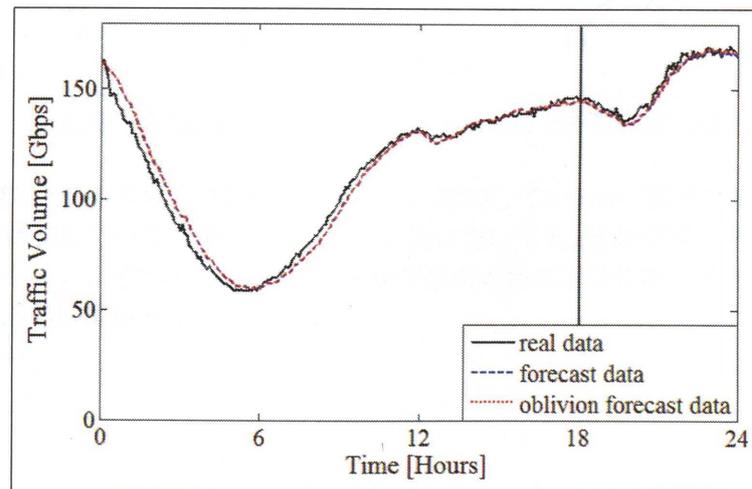


図 6.19: 18 時における逐次更新と忘却型逐次更新の比較

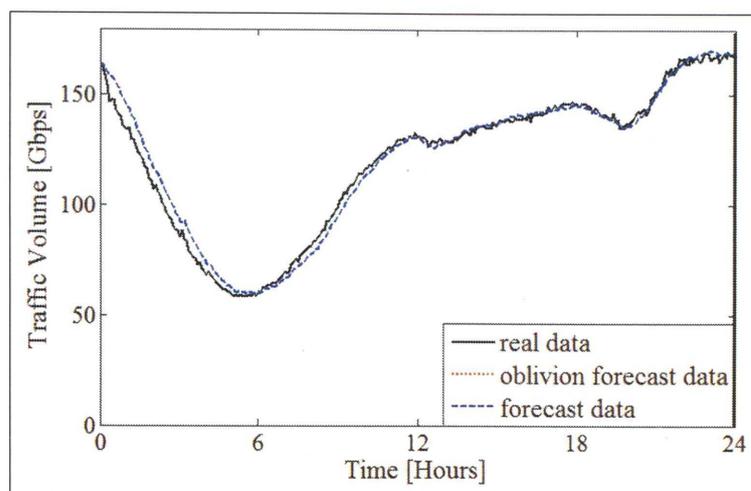


図 6.20: 24 時における逐次更新と忘却型逐次更新の比較

図 6.17, 図 6.18 を見ると, 先ほどの例と同じように, 逐次更新では未知部分に大きな誤差が出ているのに対し, 忘却型逐次更新では未知部分にそれほど大きな誤差が出ていない事が分かる.

この時の時刻と相対誤差の関係を図 6.21 に示す.

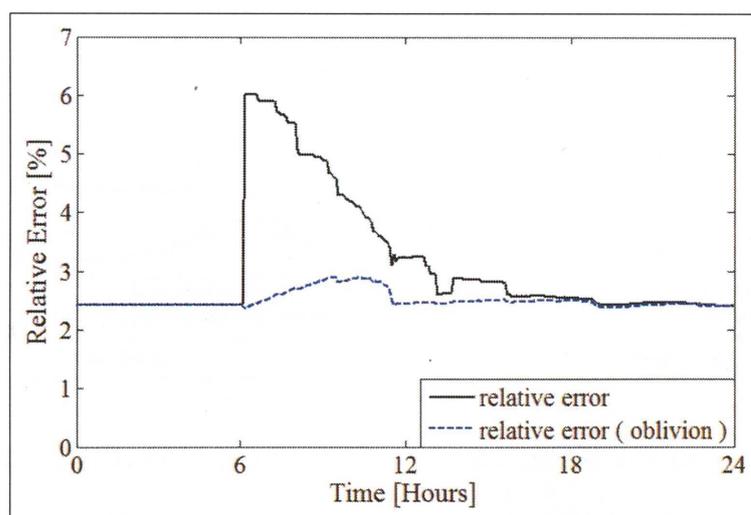


図 6.21: 逐次更新と忘却型逐次更新における相対誤差の比較

図 6.21 を見ると, 逐次更新の相対誤差が一気に増加しているが, 忘却型逐次更新の相対誤差は僅かしか増加していない事が分かる. この例においても, 忘却型逐次更新の有効性が示せていると分かる. 2 例だけの比較では, 忘却型逐次更新の有効性を示せないなので, 30 日分の時刻と相対誤差の関係を示す.

## 6.5 忘却型逐次更新の有効性検証

この章では、忘却型逐次更新の有効性を示すために30日分の予測トラヒックに対して適用し、時刻と相対誤差のグラフを示す。2012年5月10日(木)～2012年6月8日(金)の予測トラヒックに対して忘却型逐次更新と逐次更新を適用した時の、時刻と相対誤差の関係を表すグラフを図6.22～図6.53に示す。

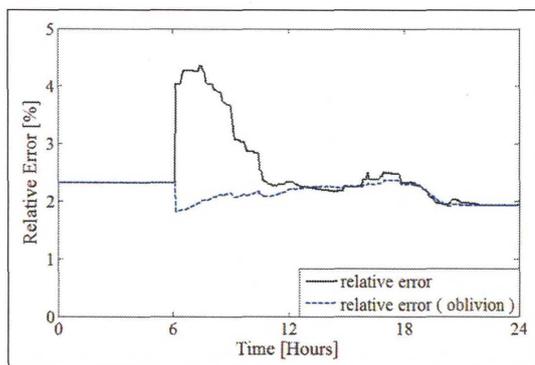


図 6.22: 2012年5月10日(木)の予測結果

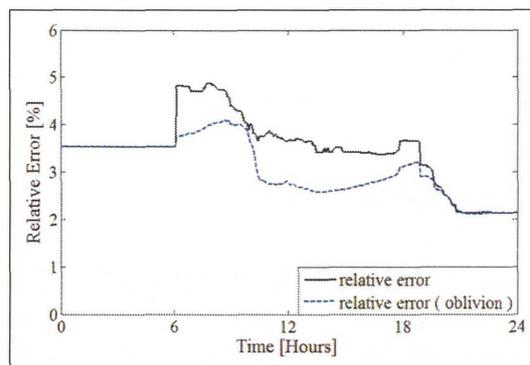


図 6.23: 2012年5月11日(金)の予測結果

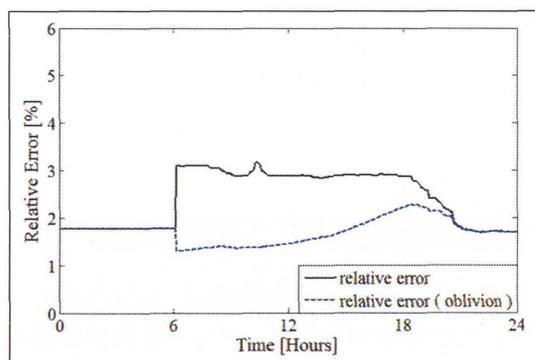


図 6.24: 2012年5月12日(土)の予測結果

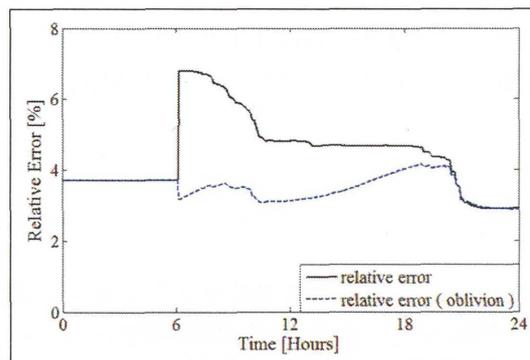


図 6.25: 2012年5月13日(日)の予測結果

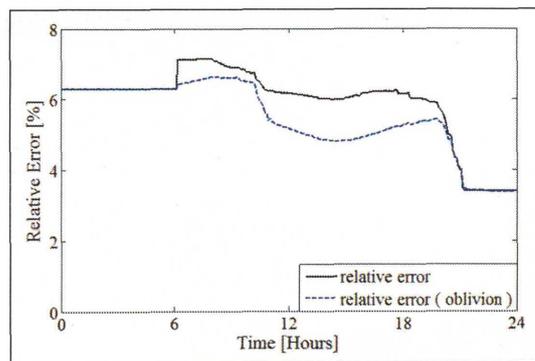


図 6.26: 2012年5月14日(月)の予測結果

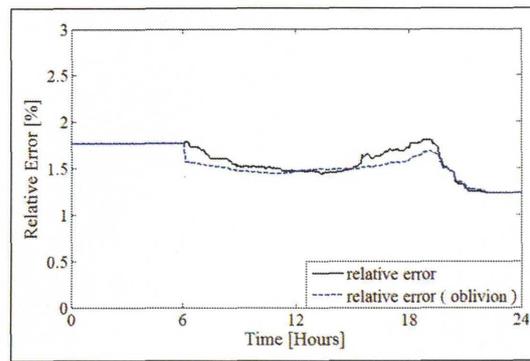


図 6.27: 2012年5月15日(火)の予測結果

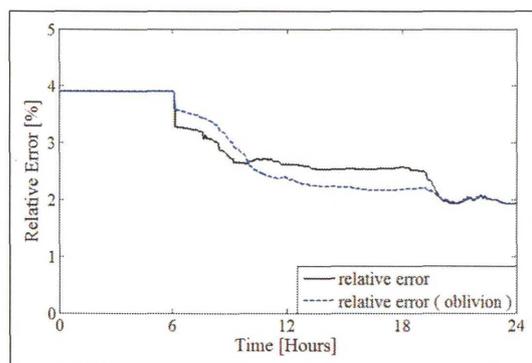
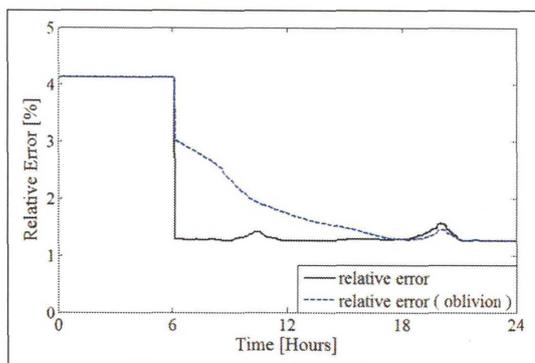


図 6.28: 2012 年 5 月 16 日 (水) の予測結果      図 6.29: 2012 年 5 月 17 日 (木) の予測結果

図 6.22 や図 6.23, 図 6.24 などの画像を見ると, 忘却型逐次更新の方が優れていると読み取れる. しかし, 逐次更新によって相対誤差が急激に落ちるパターンにおいて, 例えば図 6.28 において, 逐次更新と忘却型逐次更新の相対誤差を比べると, 逐次更新の方が急激に相対誤差が小さくなっており, 忘却型逐次更新の方は緩やかにしか相対誤差を小さくできていない事が分かる.

この事より, 逐次更新の方が忘却型逐次更新より優れている場合もあり, どちらが優れている手法であるかと断言する事は難しい. どちらがより優れた手法であるかと断言は出来ないが, 次の章で, 忘却型逐次更新と逐次更新のさらに長期における比較をし, それぞれの特徴を明らかにする.

## 6.6 逐次更新と忘却型逐次更新の性質

この章では, 忘却型逐次更新と逐次更新を多数の予測トラヒックに対して適用し, その時の性質を見ることで, それぞれの手法が本来持っている性質を明らかにする. 今回の研究では, 2012 年 4 月 7 日 (土)~2012 年 12 月 10 日 (月) までの 247 日分の逐次更新と忘却型逐次更新における統計的性質を示す.

図 6.52 に 247 日分における逐次更新の相対誤差の平均を取ったものと, 247 日分における忘却型逐次更新の相対誤差の平均を取ったものを重ねて示す.

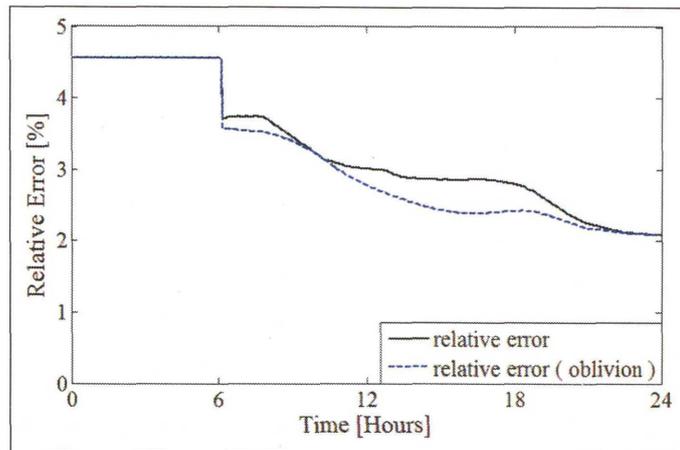


図 6.30: 逐次更新と忘却型逐次更新 (247 日分の平均)

図 6.30 において実線は逐次更新を、破線は忘却型逐次更新を表している。図 6.30 を見ると、更新を開始する前では平均 4.5% の相対誤差があると分かり、更新を開始した直後に約 1% の相対誤差を減少させる事が出来ていると分かる。また、1 日の終わりである 24 時を見ると、忘却型逐次更新・逐次更新共にほぼ同じ相対誤差になっており約 2% である事が分かる。

図 6.30 において忘却型逐次更新と逐次更新を比較すると、247 日分の平均に関しては忘却型逐次更新の方が優れていると読み取れる。次に、247 日分における忘却型逐次更新の特徴を表 6.1 に、247 日分における逐次更新の特徴を表 6.2 に示す。

表 6.1: 247 日分の忘却型逐次更新の特徴

各項目	
全日数	247 日
更新開始直後に相対誤差が減少した日数	223 日
更新開始直後に相対誤差が増加した日数	24 日
更新開始直後に相対誤差が減少する確率	90.28 %
更新開始直後における相対誤差変動量の平均	-0.993 %

表 6.2: 247 日分の逐次更新の特徴

各項目	
全日数	247 日
更新開始直後に相対誤差が減少した日数	140 日
更新開始直後に相対誤差が増加した日数	107 日
更新開始直後に相対誤差が減少する確率	56.68 %
更新開始直後における相対誤差変動量の平均	-0.858 %

表 6.1 と表 6.2 におけるそれぞれの項目の示す意味を説明する。『全日程』は、適用したすべての日数を示しており、前述の通り 247 日である。『更新開始直後に相対誤差が減少した日数』は、0 時～6 時までのトラヒックデータが得られ更新を開始した直後に、相対誤差が減少に転じた日数をカウントしたものである。『更新開始直後に相対誤差が増加した日数』は、0 時～6 時までのトラヒックデータが得られ更新を開始した直後に、相対誤差が増加に転じ

た日数をカウントしたものである。『更新開始直後に相対誤差が減少する確率』は、0時～6時までのトラヒックデータが得られ更新を開始した直後に、相対誤差が減少に転じる確率である。『更新開始直後における相対誤差変動量の平均』は、0時～6時までのトラヒックデータが得られ更新を開始した直後の、相対誤差の変動量の平均を取ったものである。

表 6.1, 表 6.2 において特に注目すべき点は、『更新開始直後に相対誤差が減少する確率』である。忘却型逐次更新においては 90.28% であり、更新を開始した直後に 90%以上の確率で相対誤差が減少すると分かる。一方、逐次更新においては 56.68% であり、更新を開始した直後には 56%の確率でしか相対誤差を減少させることができないと分かる。この項目においては、忘却型逐次更新の方が遥かに優れているという結果が得られたと分かる。

また、『更新開始直後における相対誤差変動量の平均』を見ても、忘却型逐次更新においては -0.993% であり、更新を開始した直後に平均で 0.993% の相対誤差を減少させる事が出来ていると分かる。一方、逐次更新においては -0.858% であり、更新を開始した直後には平均で 0.858% の相対誤差を減少させる事が出来ていると分かる。この項目においても、僅かではあるが忘却型逐次更新の方が優れているという結果が得られた。

表 6.1, 表 6.2 においては忘却型逐次更新の方が優れているという結果が出たが、『忘却型逐次更新』と『逐次更新』は一長一短であり、どちらが優れているとは言えないと我々は考える。しかし、無難で妥当な予測結果が得たいと考えているならば、『忘却型逐次更新』の方が優れていると言える。

## 第7章 大規模通信障害

この第7章では、大規模通信障害が発生してトラフィック量に大きな変動が見られた時、それをどのように検知するのか、また障害発生後のトラフィック量をどのように予測するのかについて説明する。本研究では JPIX のトラフィック量に大きな変動が見られた 2012 年 3 月 11 日 (金) の東日本大震災当日を例として解説していく。

### 7.1 東日本大震災当日における JPIX のトラフィック

東日本大震災当日における JPIX のトラフィックデータを図 7.1 に示す。

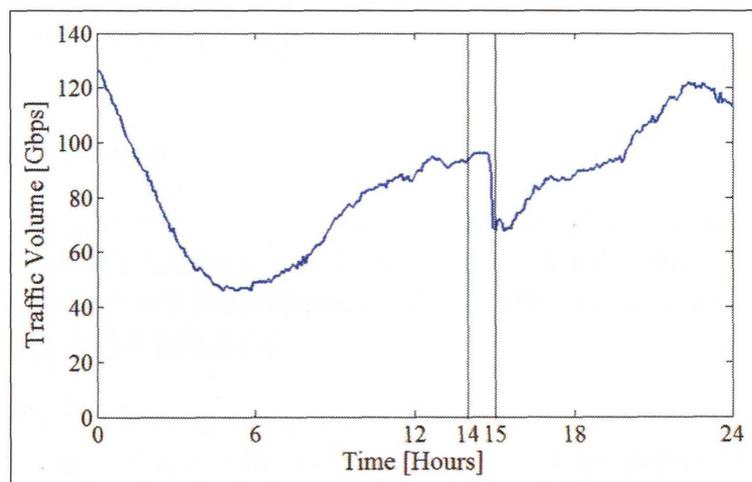


図 7.1: 震災当日における JPIX のトラフィック

図 7.1 を見ると、14 時～15 時の間にトラフィックに大きな変動が見られていると分かる。この部分を拡大したものを図 7.2 に示す。

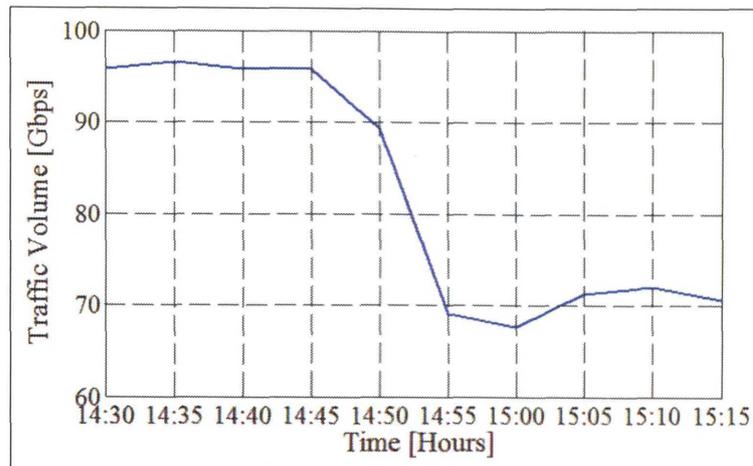


図 7.2: 震災発生時における JPIX のトラフィックの変動

図 7.2 を見ると、14 時 45 分時点では 95.80Gbps のトラフィック量があるのに対して、その 10 分後の 14 時 55 分時点では 69.15Gbps のトラフィック量しか無い事が分かる。これは、東日本大震災が発生した時刻 14 時 46 分 18 秒 と一致する。この事からも、この大規模通信障害の原因が東日本大震災であると分かる。

## 7.2 通信障害の検知

この章では、通信障害が発生した事を検知する方法を説明する。大規模通信障害が発生するとトラフィック量に大きな変動が見られるので、その大きな変動を検知して通信障害の発生を検知する。トラフィックの大きな変動を検知するには様々な方法が考えられるが、我々は『逐次更新』を用いた方法を提案する。

逐次更新を行っている場合において、5 分後に得られるトラフィックデータは比較的高精度で予測する事が出来る。例えば、0 時～10 時までのトラフィックデータを用いて逐次更新を行った場合、10 時 5 分に得られる 1 点分のトラフィックデータは高精度で予測が可能である。1 日のトラフィックを予測する場合においては 6 時から逐次更新を開始したが、通信障害の検知を行う場合において逐次更新は 0 時からスタートする。

例として、2012 年 3 月 11 日 (日)～2012 年 4 月 7 日 (土) までの 28 日分のトラフィックを用いて 2012 年 4 月 8 日 (日) の予測トラフィックを作成し、0 時から逐次更新を開始する。この時の、逐次更新における 5 分後の予測トラフィックと、実際に 5 分後に得られたトラフィックの誤差を各時刻で求めた時のグラフを図 7.3 に示す。

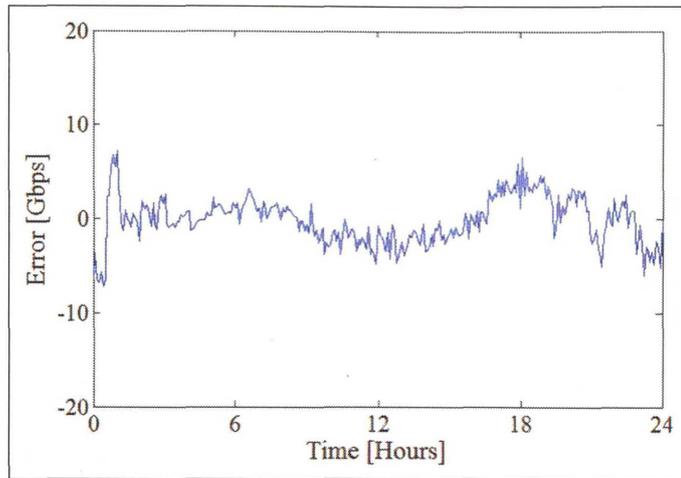


図 7.3: 2012 年 4 月 8 日 (日) の逐次更新における 5 分後の誤差

図 7.3 を見ると、トラフィック量の誤差は 10Gbps 以下になっており、極端に大きな誤差は見られない。

これに対し、東日本大震災当日においては大きな誤差が観測された。2011 年 2 月 4 日 (金) ~2011 年 3 月 3 日 (木) までの 28 日分のトラフィックを用いて 2012 年 4 月 8 日 (日) の予測トラフィックを作成し、0 時から逐次更新を開始する。この時の、逐次更新における 5 分後の予測トラフィックと、実際に 5 分後に得られたトラフィックの誤差を各時刻で求めた時のグラフを図 7.4 に示す。

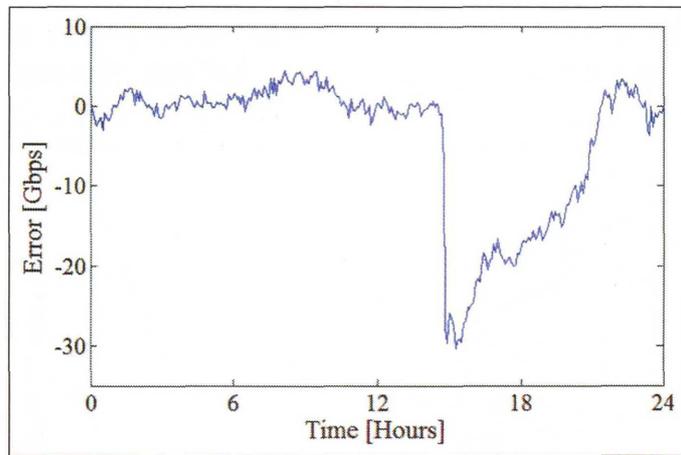


図 7.4: 東日本大震災当日の逐次更新における 5 分後の誤差

図 7.4 を見ると、東日本大震災が発生した 2 時 46 分頃から大きな誤差が観測されていると分かる。最大で 30Gbps 以上の誤差が観測され、明らかにトラフィック量に異常が見られたと分かる。この時、閾値を設定すれば自動で異常検出が可能である。例えば、閾値を 20Gbps と設定すれば今回の通信障害を検知する事が出来る。以上が、通信障害を検知する為の方法である。

### 7.3 通信障害発生時における逐次更新

第7.2章では、通信障害を検知する方法について説明したが、この第7.3章では通信障害が発生した後のトラフィックをどのように予測するかについて説明する。

通信障害が発生後のトラフィックの予測法には、単純な方法を用いる。通信障害が発生したと検知された場合、それまで逐次更新を行っていた部分を切り離し、大きく変動した点に逐次更新のトラフィック量を合わせ、その後逐次更新を再開する。この事を説明すると図7.5～7.10のようになる。

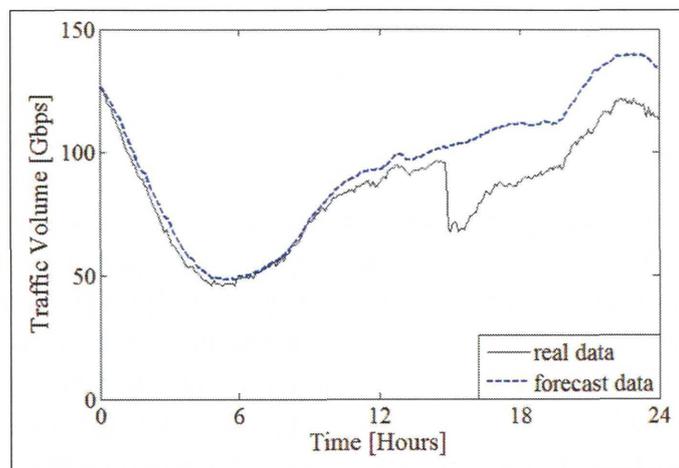


図 7.5: 東日本大震災当日の予測トラフィックと実際のトラフィック (更新前)

図7.5は逐次更新を開始する前の予測トラフィックと実際のトラフィックである。まだ通信障害の事を検知する前なので、通信障害が発生した後に大きな誤差が生じている。

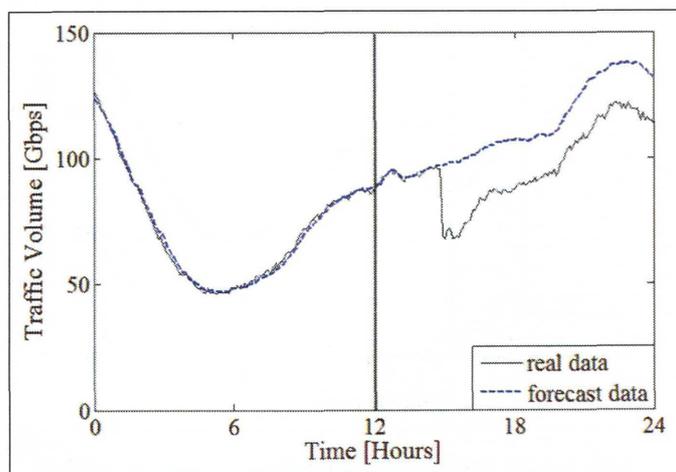


図 7.6: 東日本大震災当日の12時における更新

図7.6は12時における逐次更新トラフィックと実際のトラフィックである。東日本大震災発生前までは良い予測ができていたが、東日本大震災を予測できていないので14時45分以降に大きな誤差が出ている。

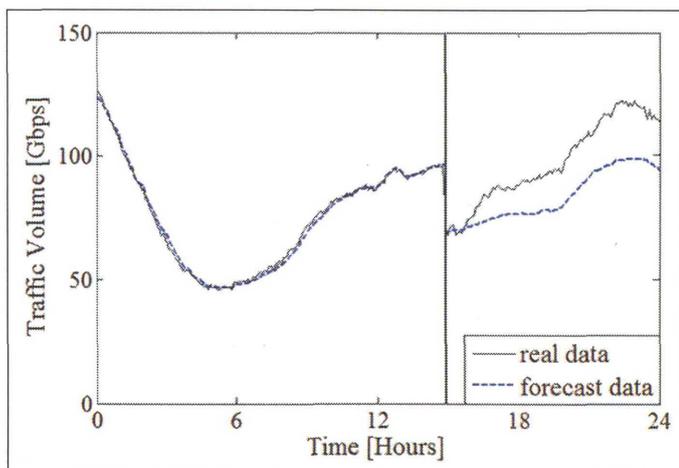


図 7.7: 東日本大震災当日における通信障害検知直後

図 7.7 は通信障害を検知した直後の逐次更新トラフィックと実際のトラフィックである。通信障害を検知すると、それ以前の逐次更新を切り離す。図 7.7 を見ると、通信障害発生前と後で予測トラフィックが切り離された状態になっているが、このように以前の逐次更新はそのまま、通信障害発生後から新たに逐次更新を開始する。通信障害が発生するとトラフィック量が大幅に変動するので、障害発生直後に逐次更新トラフィックと実際のトラフィックのボリュームを合わせる。東日本大震災においては 14 時 55 分のトラフィック量が 69.15Gbps に減少したので、逐次更新の 14 時 55 分のトラフィック量も 69.15Gbps になるようにトラフィック全体に同じ値を掛け合わせ、ボリュームの合わせ込みを行う。ボリュームを合わせ込む方法としては、 $u_1, u_2$  に同じ値を掛け合わせ、ボリュームを変動させるという方法を取っている。図 7.7 に示したグラフは、ボリューム合わせを行った時の逐次更新を表している。

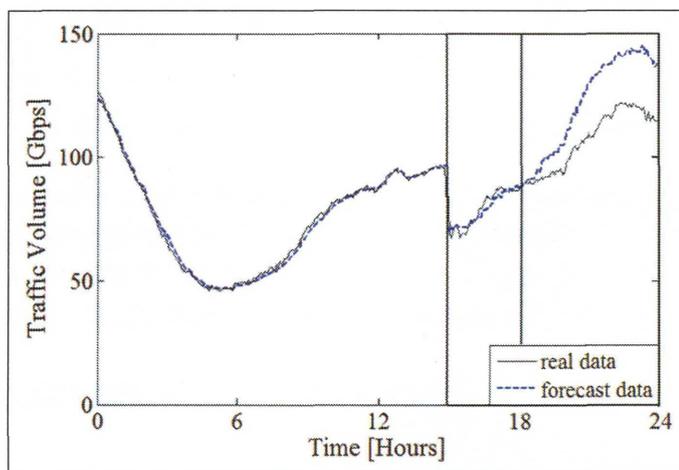


図 7.8: 東日本大震災当日の 18 時における更新

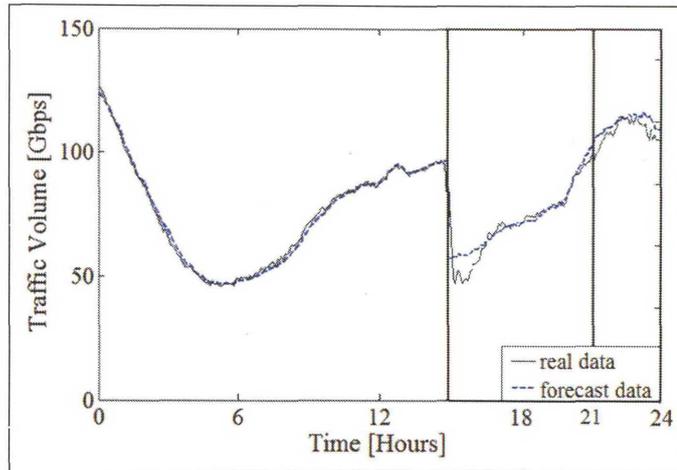


図 7.9: 東日本大震災当日の 21 時における更新

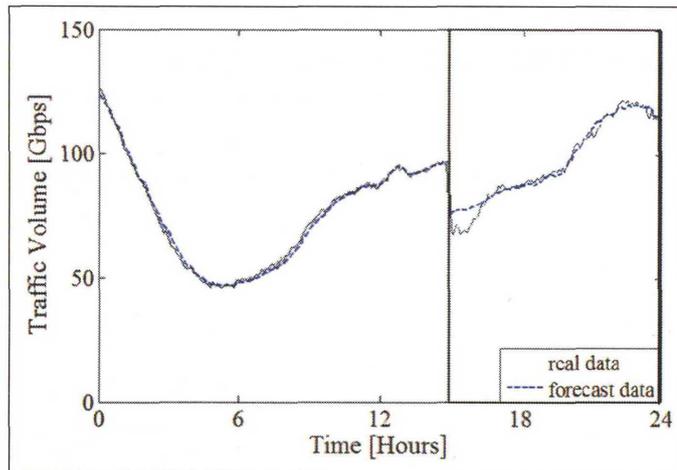


図 7.10: 東日本大震災当日の 24 時における更新

図 7.8～図 7.10 を見ると、データが得られるほど徐々に予測の精度が上がっている事が分かり、図 7.10 では逐次更新の波形と実際のトラフィックの波形がほぼ一致している事が分かる。この事より、東日本大震災当日のようなイレギュラーな場合であっても、第 1 成分と第 2 成分の組み合わせによって高い精度でトラフィックが再現できると分かる。

また、この時の逐次更新における 5 分後の予測トラフィックと実際のトラフィックの誤差を図 7.11 に示す。

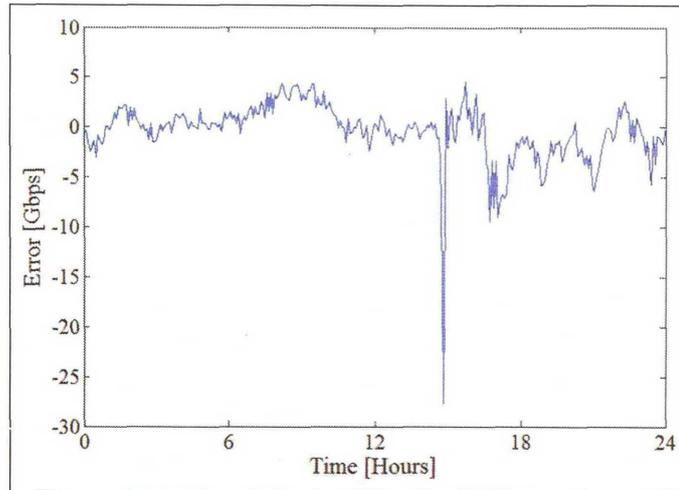


図 7.11: 逐次更新における 5 分後の予測トラヒックと実際のトラヒックの誤差

図 7.11 を見ると、震災発生直後には 25Gbps 以上の非常に大きな誤差が出ているが、それ以降は 10Gbps 以上の誤差は観測されていない。図 7.11 を図 7.4 と比較すると、図 7.11 の方が誤差が少なくこの方法の優位性が示せている。

以上が大規模通信障害発生時におけるトラヒックの予測法である。

## 第8章 結論

本研究では主成分分析手法である特異値分解を用いて JPIX トラフィックに含まれる主成分を抽出し、その主成分を用いる事で将来のトラフィックを予測した。また、予測日当日に得られる5分に1点のデータを用いる事で、前日までに作成した予測トラフィックを逐次的に更新するという手法の提案を行った。結果を簡単に述べると、前日までに作成した予測は平均相対誤差 4.5% 程度、そこから更新をしていく事で平均相対誤差を 2% 程度まで減少させることが可能であるという結果が得られた。

本手法では、予測データの逐次更新を行う事で高い精度の予測を可能とし、通信障害などに伴う突発的なトラフィック変動を迅速に検知する事ができ、通信障害発生後のトラフィック量についても予測する事を可能にした。本稿では東日本大震災の例しか取り上げなかったが、通信障害が発生した他の例についても同様の手法を適用し、手法の有効性を検証していきたい。

今後の展望として、海外のバックボーンネットワークである LONAP (London-based Internet Exchange Point) や NYIIX(New York International Internet eXchange)などを対象として解析する事で本手法の有用性を示したい。また、現在スマートフォンやタブレット型端末などのモバイル端末が増加しており、モバイル端末によるトラフィックが急増している。一般的に、モバイル端末によるトラフィックは固定端末によるトラフィックより変動が大きく、予測が困難と言われている。今後もし、モバイル端末のトラフィックデータを入手する事ができれば、モバイル端末のトラフィックについて予測を行う手法の提案も行いたい。

## 謝辞

本修士論文は、筆者が法政大学大学院 工学研究科 情報電子工学専攻 修士課程在学中に情報処理工学研究室において行った研究をまとめたものです。本研究を進めるに際して、終始熱心なご指導を頂いた八名和夫教授、濱田寛久氏、鈴木悠太氏並びに諸先輩方、共同研究者としてご協力いただいた山下雅人氏、そして八名研究室の皆様に御礼申し上げます。

## 参考文献

- [1] 総務省情報通信データベース (<http://www.soumu.go.jp/johotusintokei/>)
- [2] 宗像誠之, "NTT の自縛," pp. 18-32, 日経コミュニケーション, 2008.
- [3] JPIX : *JapanInternetExchange* (<http://www.jpix.ad.jp>)
- [4] 濱田寛久 "特異値分解を用いた JPIX トラヒックの適応的短期予測," 2011 年度法政大学大学院工学研究科情報電子工学専攻修士論文, pp1-4(2011-3)
- [5] 宮島崇之, 八名和夫, "状態空間モデルを用いた JPIX トラヒックの成分分解," 電子情報通信学会技術研究報告書. NS, ネットワークシステム 108(286), 13-27, 11 月, 2008.
- [6] 依田浩和, 八名和夫, "状態空間モデルによる JPIX トラヒックの成分分解," 電子情報通信学会技術研究報告書. IN, 情報ネットワーク 107(249), 59-62, 10 月, 2007.
- [7] 北川源四郎, "時系列解析入門," pp. 151-177, 岩波書店, 2005.
- [8] H. Zhao, N. Ansari, and Yun Q., Shi, "Network Traffic Prediction Using Least Mean Kurtosis," IEICE Transactions on Communication, Vol. E89-B, No. 5, pp. 1672-1674, 2006.
- [9] Zhang Shen Fu-ku and Chang Pan Wei, "An Engineering Approach to Prediction of Network Traffic Based on Time-Series Model," Proc. 2009 IEEE International Joint Conference on Artificial Intelligence, pp. 432-435, 2009.
- [10] Guoqiang Mao, "A Timescale Decomposition Approach to Network Traffic Prediction," IEICE Transactions on Communication, Vol. E88-B, No.10, 2005.

# 付録A 特異値分解に用いたトラフィックデータ

2012年11月9日～2012年11月28日までのトラフィックデータを示す。

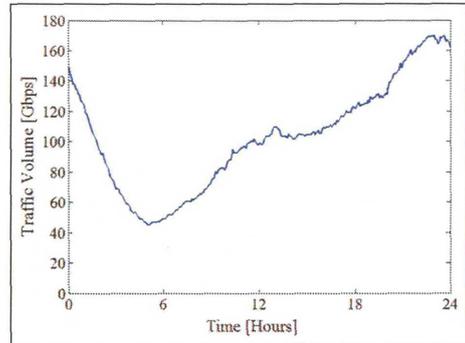
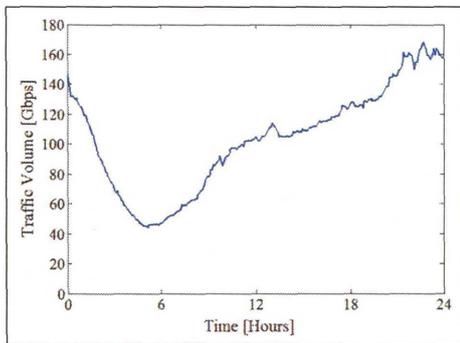


図 A.1: 2012年11月5日(月)のトラフィック 図 A.2: 2012年11月6日(火)のトラフィック

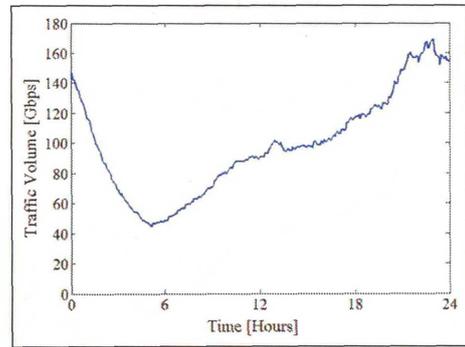
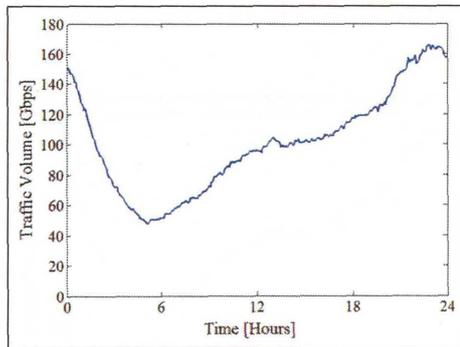


図 A.3: 2012年11月7日(水)のトラフィック 図 A.4: 2012年11月8日(木)のトラフィック

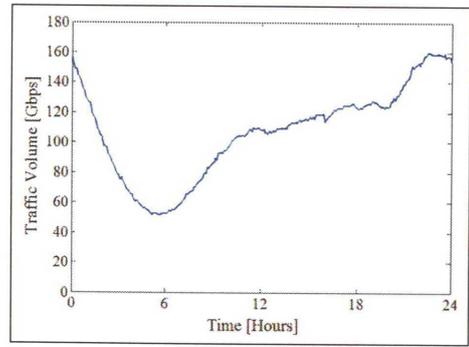
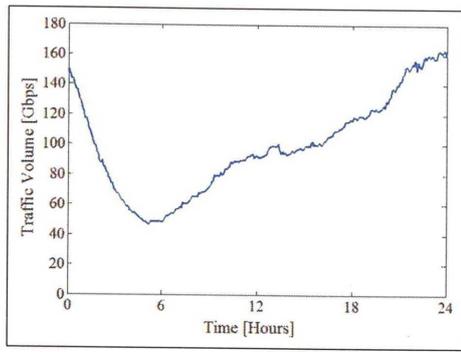


図 A.5: 2012年11月9日(金)のトラフィック 図 A.6: 2012年11月10日(土)のトラフィック

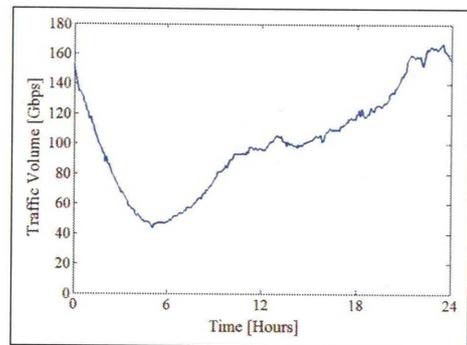
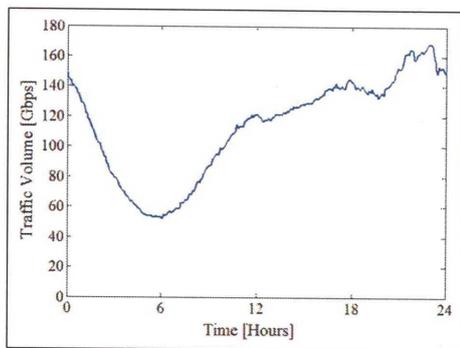


図 A.7: 2012年11月11日(日)のトラフィック 図 A.8: 2012年11月12日(月)のトラフィック

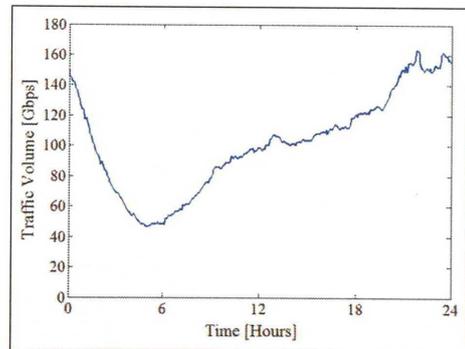
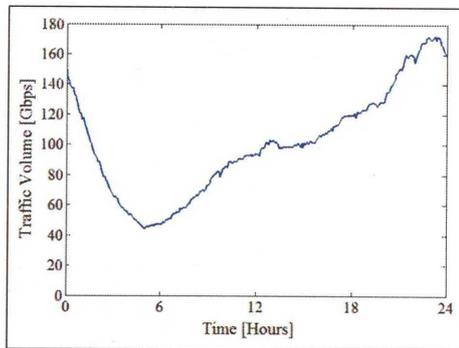


図 A.9: 2012年11月13日(火)のトラフィック 図 A.10: 2012年11月14日(水)のトラフィック

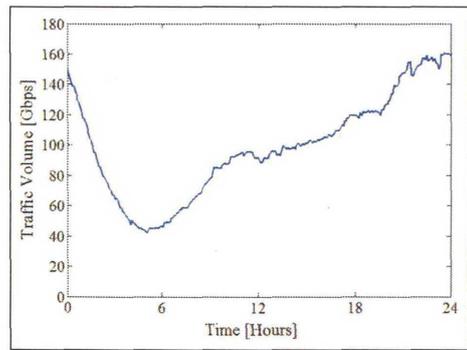
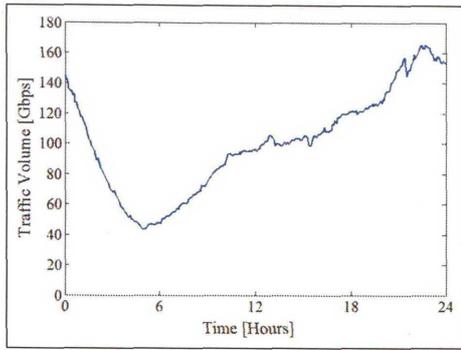


図 A.11: 2012年11月15日(木)のトラフィック 図 A.12: 2012年11月16日(金)のトラフィック

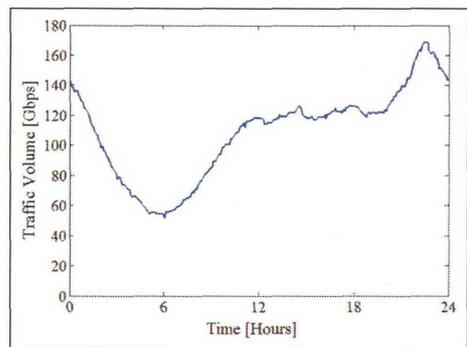
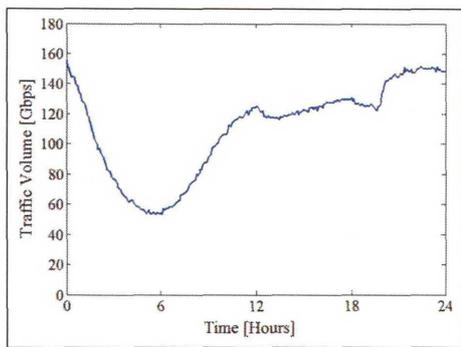


図 A.13: 2012年11月17日(土)のトラフィック 図 A.14: 2012年11月18日(日)のトラフィック

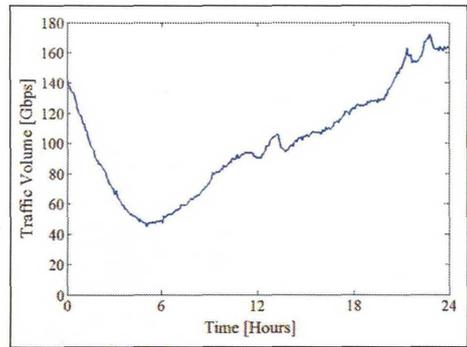
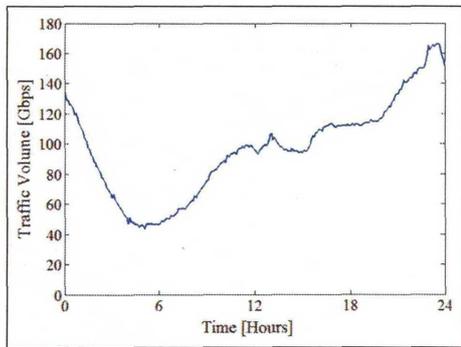


図 A.15: 2012年11月19日(月)のトラフィック 図 A.16: 2012年11月20日(火)のトラフィック

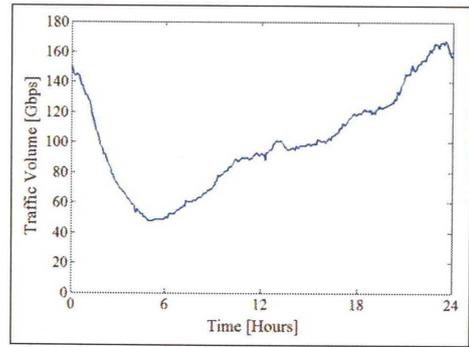
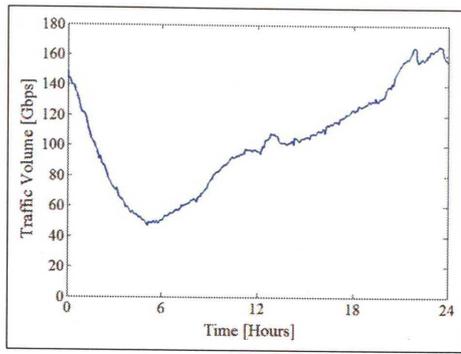


図 A.17: 2012年11月21日(水)のトラフィック 図 A.18: 2012年11月22日(木)のトラフィック

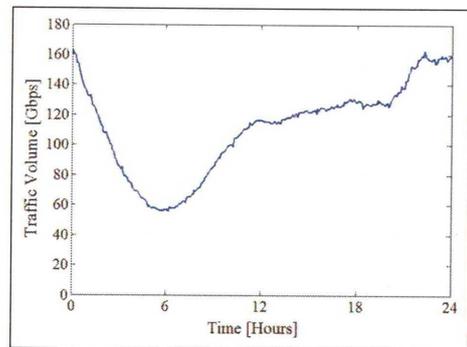
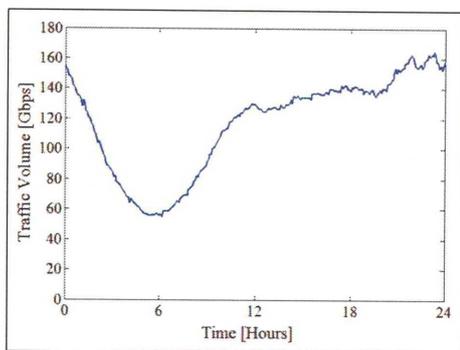


図 A.19: 2012年11月23日(金)のトラフィック 図 A.20: 2012年11月24日(土)のトラフィック

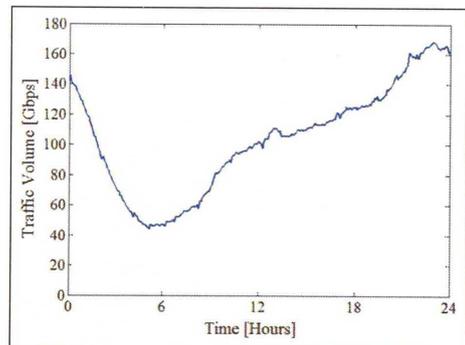
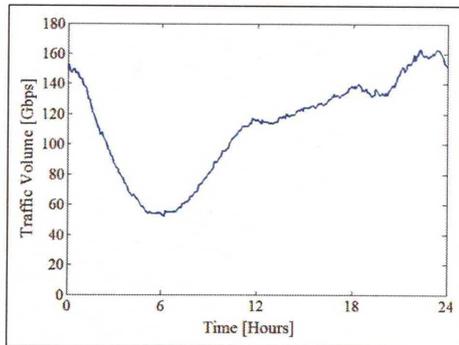


図 A.21: 2012年11月25日(日)のトラフィック 図 A.22: 2012年11月26日(月)のトラフィック

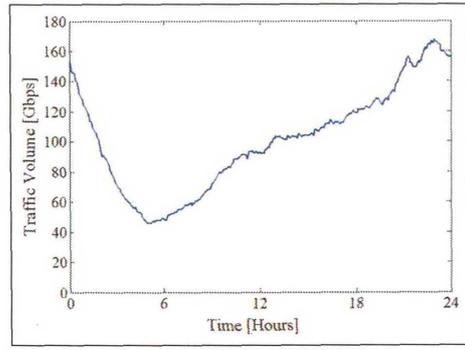
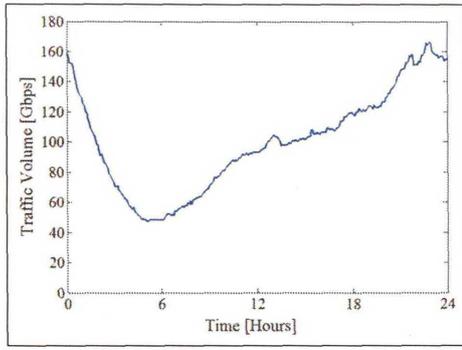


図 A.23: 2012年11月27日(火)のトラフィック 図 A.24: 2012年11月28日(水)のトラフィック

## 付録B 特徴成分

2012年11月9日～2012年11月28日までのトラヒックデータを特異値分解した時の、第5成分～第28成分を示す。

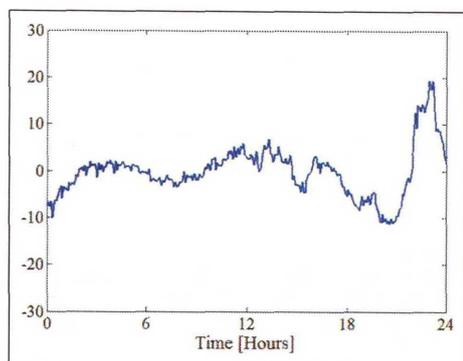


図 B.1: 第5成分

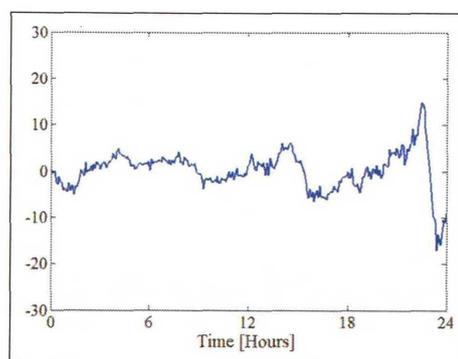


図 B.2: 第6成分

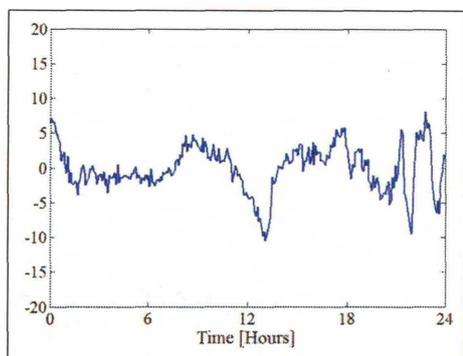


図 B.3: 第7成分

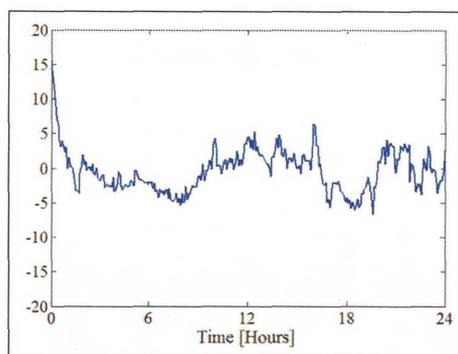


図 B.4: 第8成分

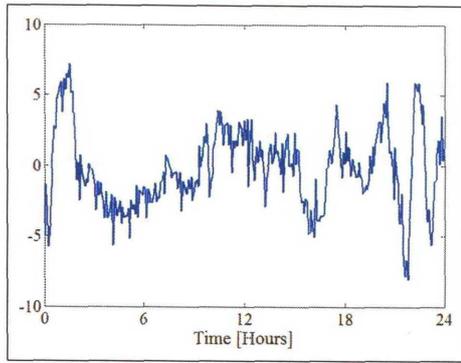


图 B.5: 第 9 成分

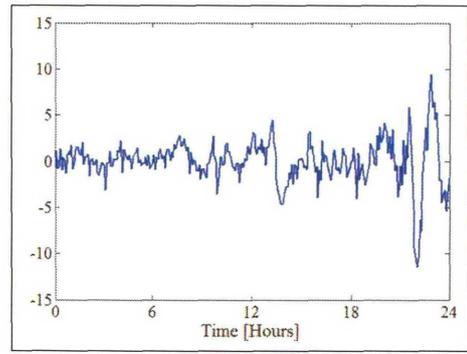


图 B.6: 第 10 成分

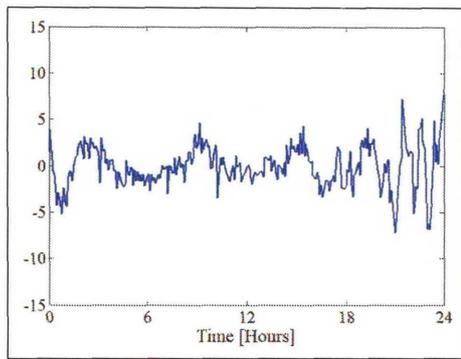


图 B.7: 第 11 成分

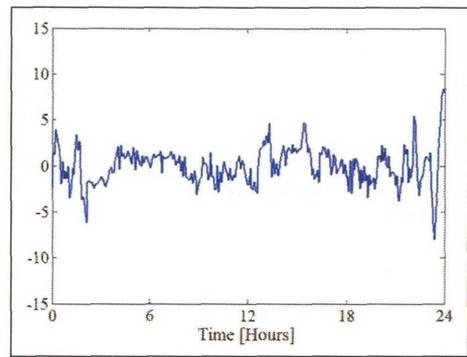


图 B.8: 第 12 成分

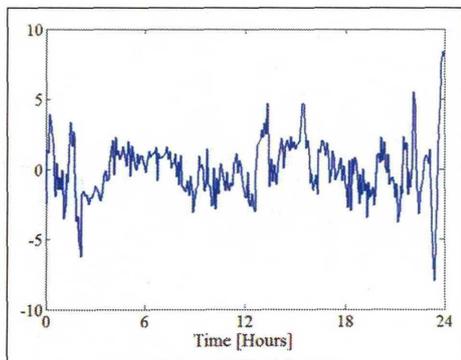


图 B.9: 第 13 成分

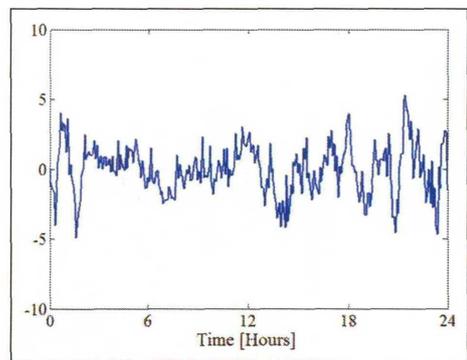


图 B.10: 第 14 成分

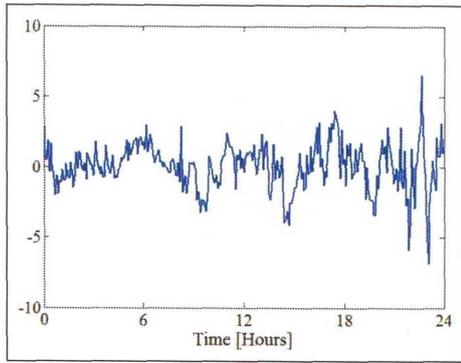


图 B.11: 第 15 成分

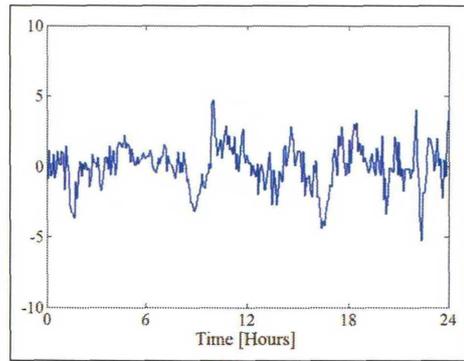


图 B.12: 第 16 成分

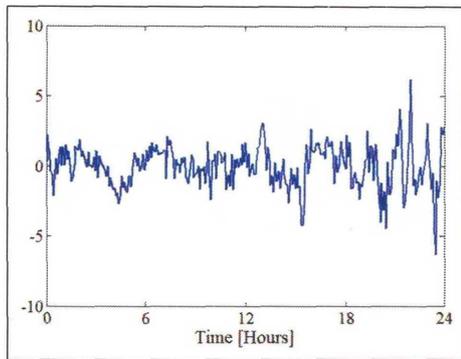


图 B.13: 第 17 成分

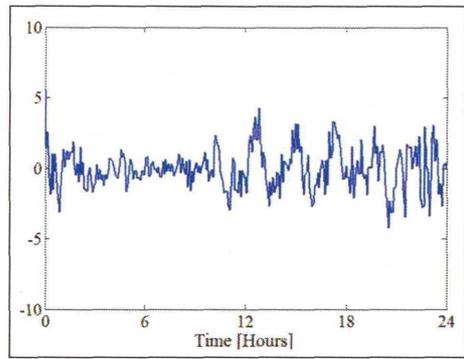


图 B.14: 第 18 成分

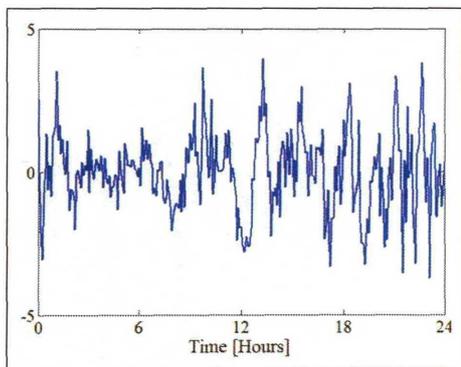


图 B.15: 第 19 成分

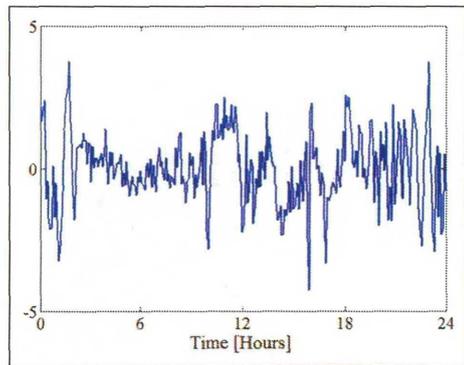


图 B.16: 第 20 成分

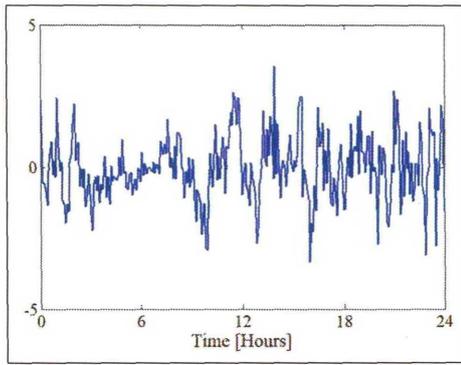


图 B.17: 第 21 成分

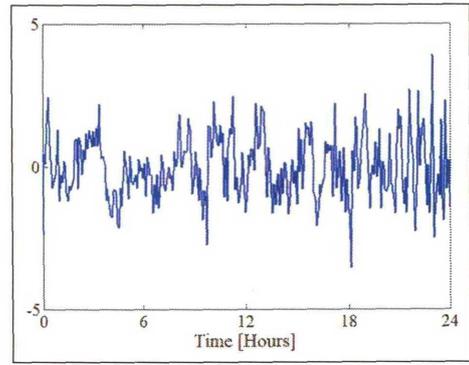


图 B.18: 第 22 成分

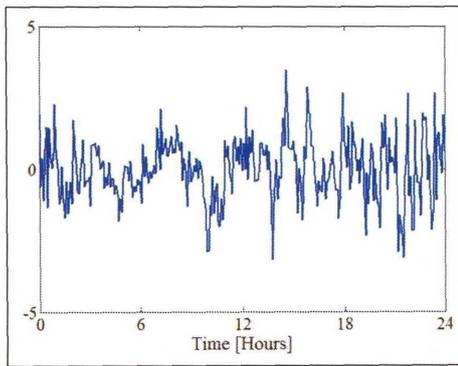


图 B.19: 第 23 成分

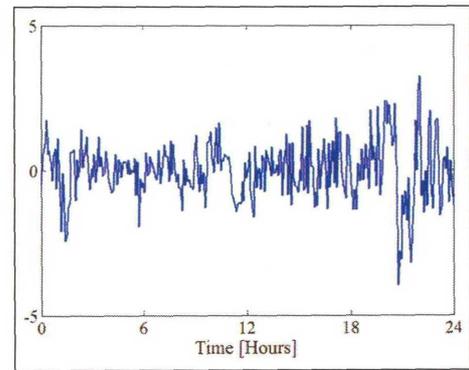


图 B.20: 第 24 成分

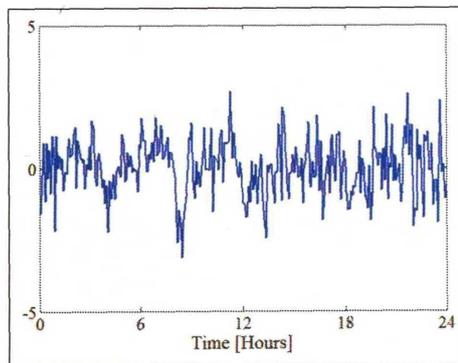


图 B.21: 第 25 成分

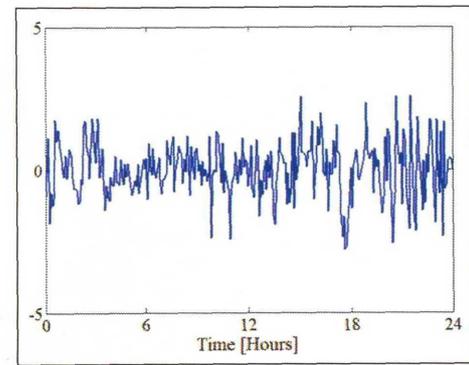


图 B.22: 第 26 成分

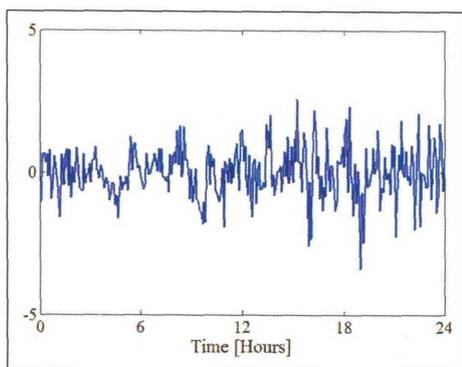


图 B.23: 第 27 成分

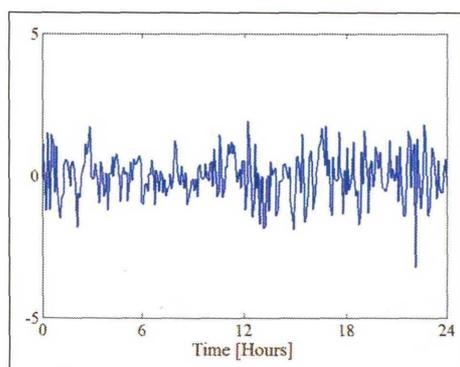


图 B.24: 第 28 成分

## 付録C 各成分に対する $U$

2012年11月9日～2012年11月28日までのトラフィックデータを特異値分解した時の、第5成分～第28成分に対する  $U$  を示す。

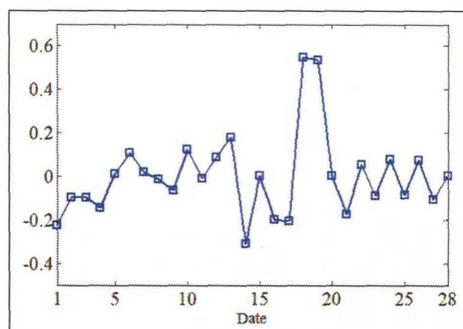


図 C.1: 第5成分に対する  $U$

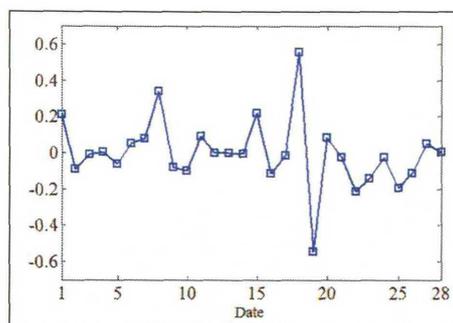


図 C.2: 第6成分に対する  $U$

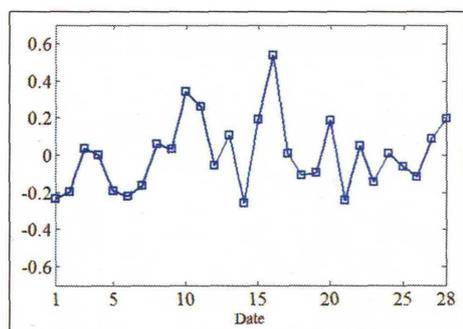


図 C.3: 第7成分に対する  $U$

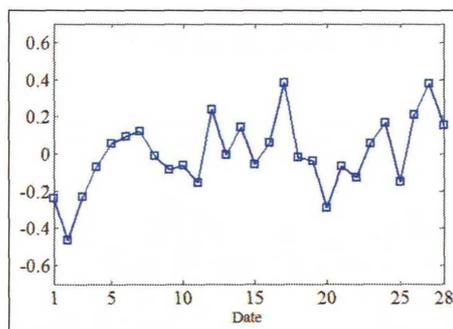


図 C.4: 第8成分に対する  $U$

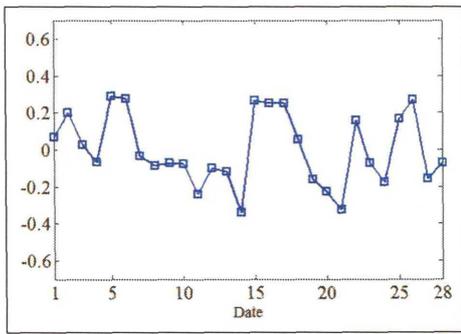


図 C.5: 第 9 成分に対する  $U$

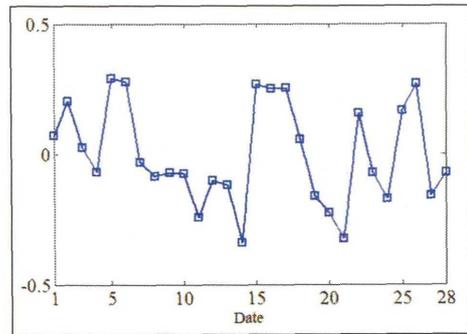


図 C.6: 第 10 成分に対する  $U$

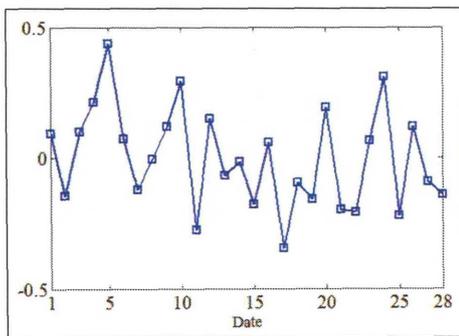


図 C.7: 第 11 成分に対する  $U$

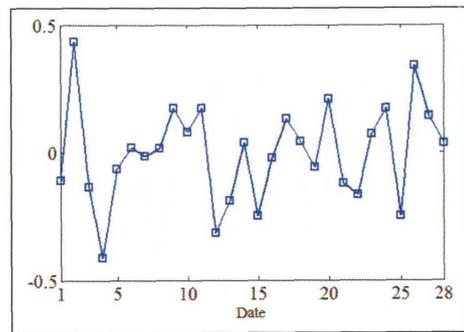


図 C.8: 第 12 成分に対する  $U$

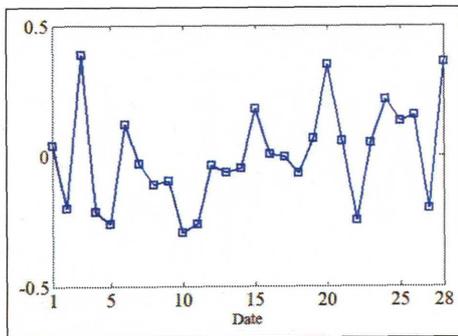


図 C.9: 第 13 成分に対する  $U$

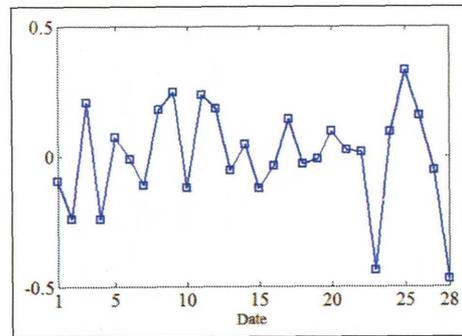


図 C.10: 第 14 成分に対する  $U$

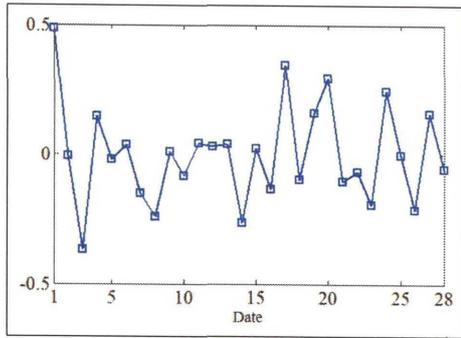


図 C.11: 第 15 成分に対する  $U$

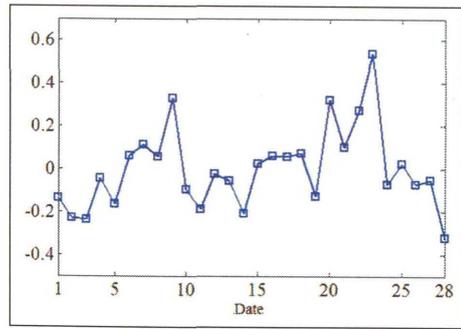


図 C.12: 第 16 成分に対する  $U$

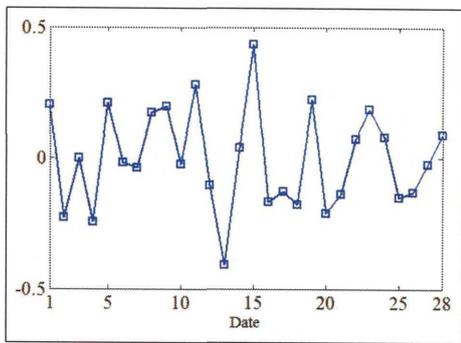


図 C.13: 第 17 成分に対する  $U$

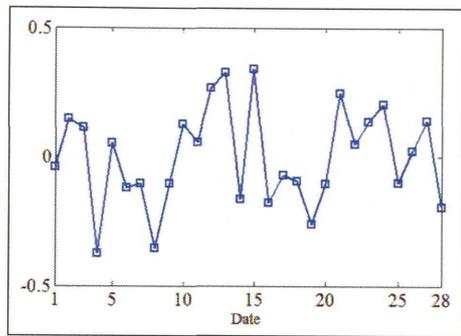


図 C.14: 第 18 成分に対する  $U$

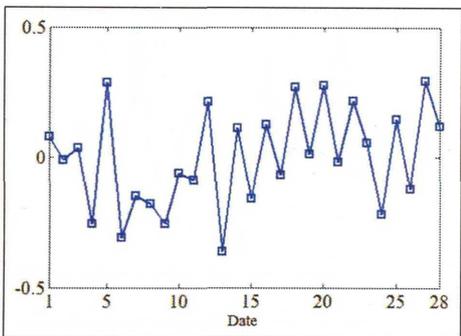


図 C.15: 第 19 成分に対する  $U$

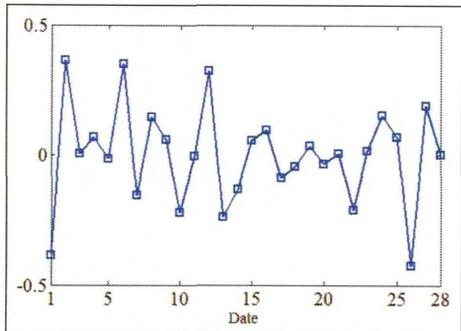


図 C.16: 第 20 成分に対する  $U$

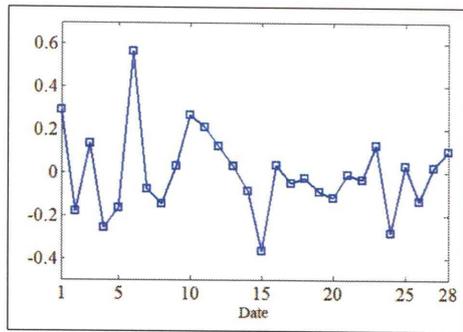


図 C.17: 第 21 成分に対する  $U$

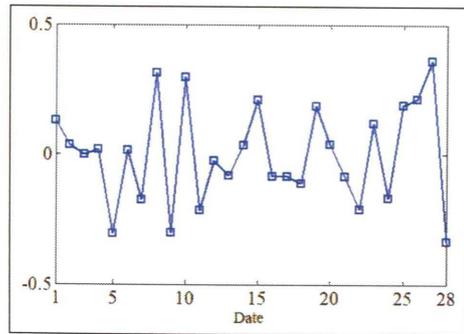


図 C.18: 第 22 成分に対する  $U$

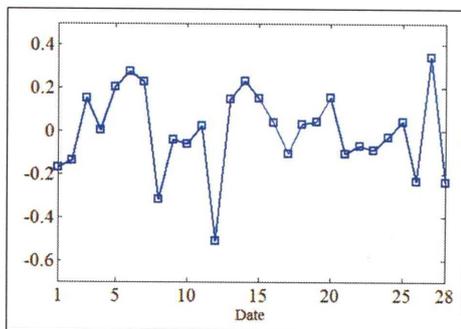


図 C.19: 第 23 成分に対する  $U$

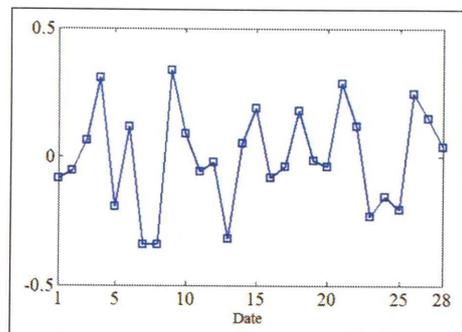


図 C.20: 第 24 成分に対する  $U$

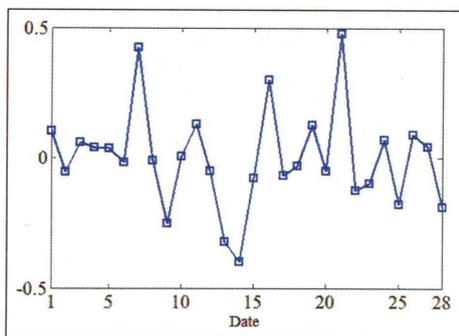


図 C.21: 第 25 成分に対する  $U$

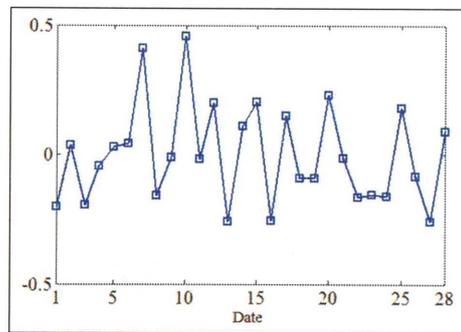


図 C.22: 第 26 成分に対する  $U$

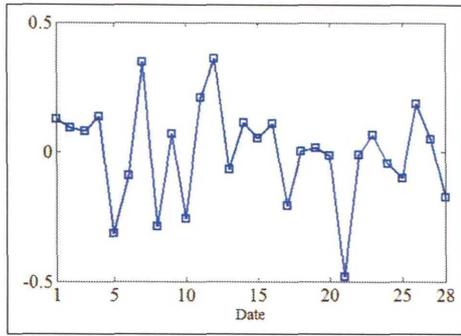


図 C.23: 第27成分に対する  $U$

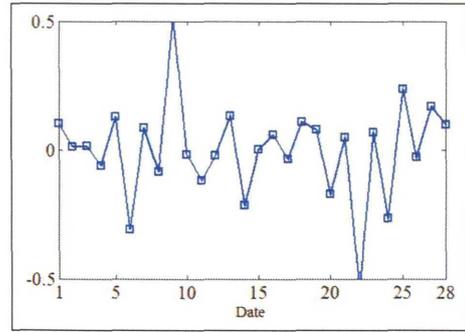


図 C.24: 第28成分に対する  $U$

## 付録D 相対誤差と時刻の関係

2012年4月16日(月)～2012年5月9日(水)の予測データに対して逐次更新を行った時の相対誤差と時刻の関係を示す。

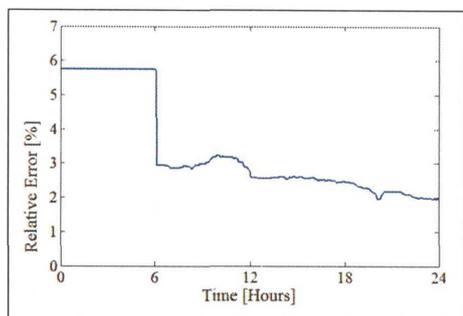


図 D.1: 2012年4月16日(月)の相対誤差

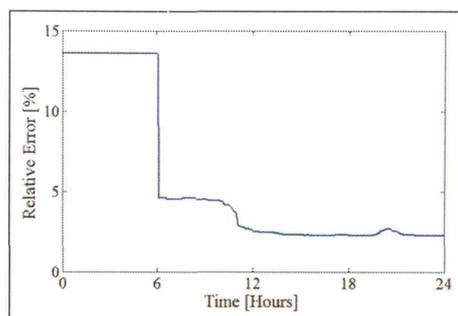


図 D.2: 2012年4月17日(火)の相対誤差

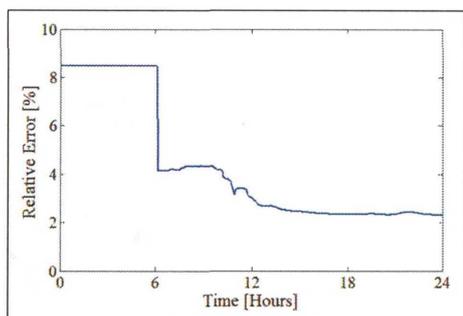


図 D.3: 2012年4月18日(水)の相対誤差

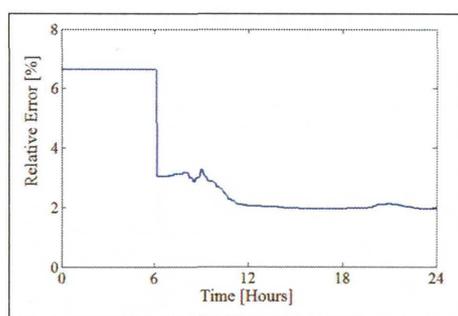


図 D.4: 2012年4月19日(木)の相対誤差

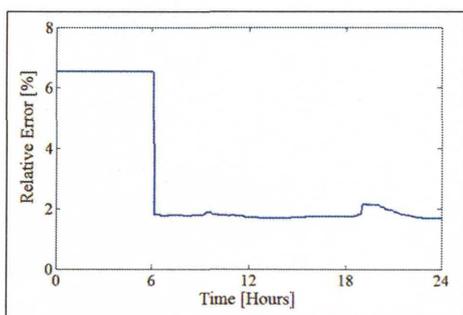


図 D.5: 2012年4月20日(金)の相対誤差

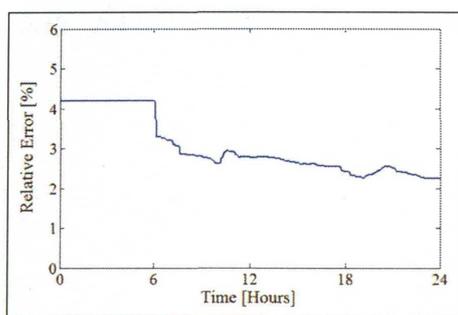


図 D.6: 2012年4月21日(土)の相対誤差

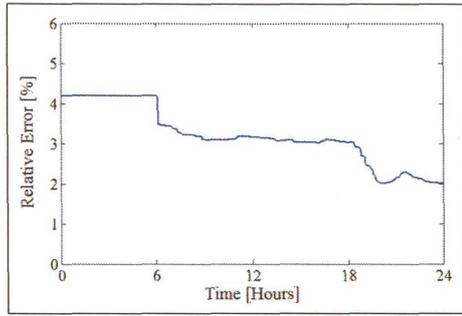


図 D.7: 2012年4月22日(日)の相対誤差

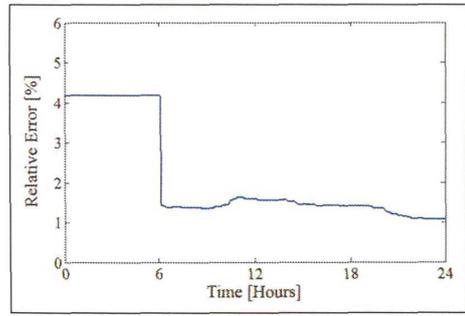


図 D.8: 2012年4月23日(月)の相対誤差

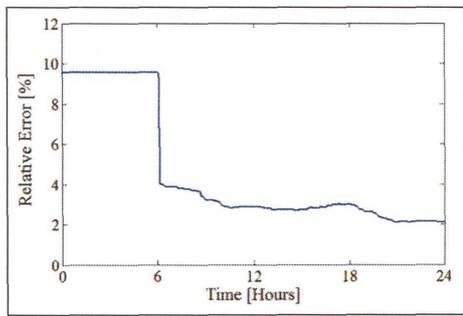


図 D.9: 2012年4月24日(火)の相対誤差

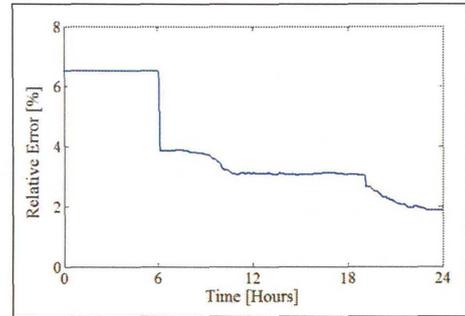


図 D.10: 2012年4月25日(水)の相対誤差

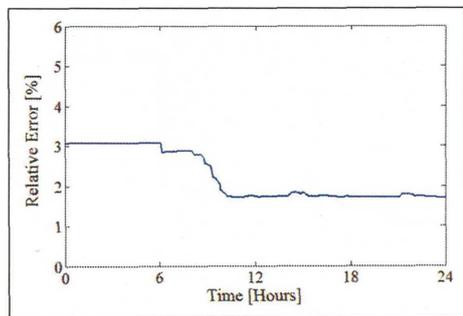


図 D.11: 2012年4月26日(木)の相対誤差

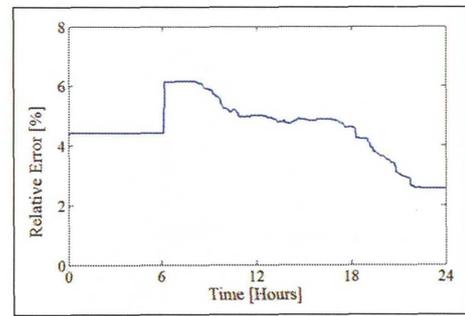


図 D.12: 2012年4月27日(金)の相対誤差

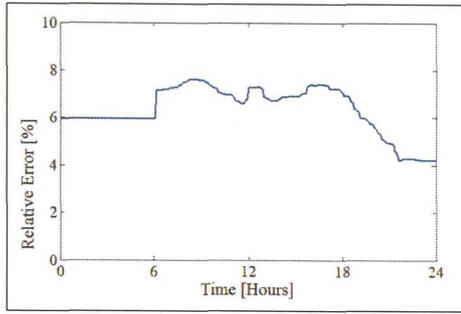


図 D.13: 2012年4月28日(土)の相対誤差

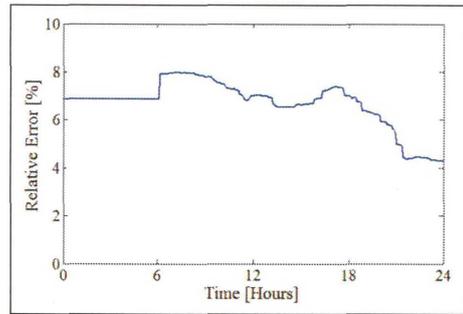


図 D.14: 2012年4月29日(日)の相対誤差

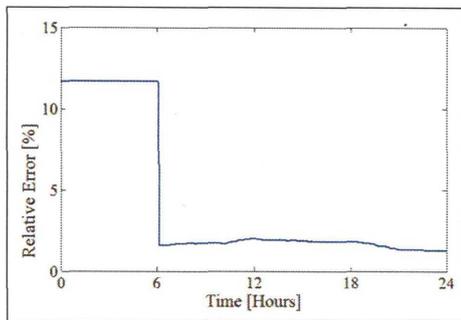


図 D.15: 2012年4月30日(月)の相対誤差

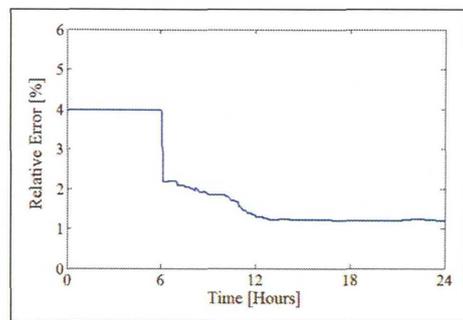


図 D.16: 2012年5月1日(火)の相対誤差

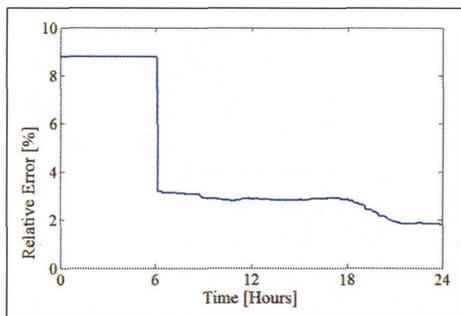


図 D.17: 2012年5月2日(水)の相対誤差

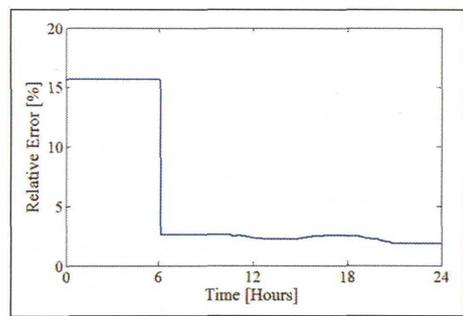


図 D.18: 2012年5月3日(木)の相対誤差

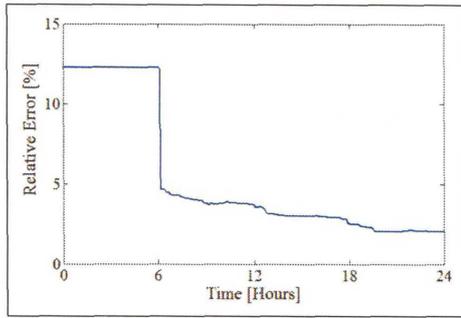


図 D.19: 2012年5月4日(金)の相対誤差

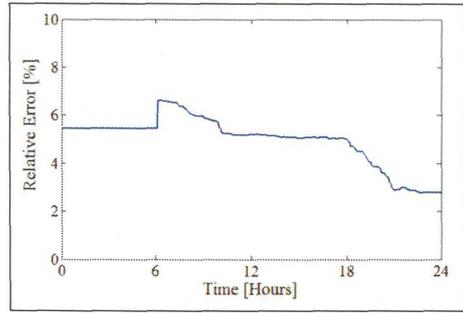


図 D.20: 2012年5月5日(土)の相対誤差

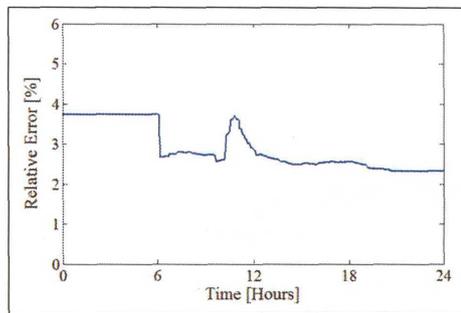


図 D.21: 2012年5月6日(日)の相対誤差

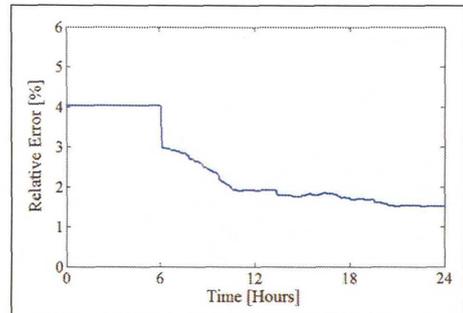


図 D.22: 2012年5月7日(月)の相対誤差

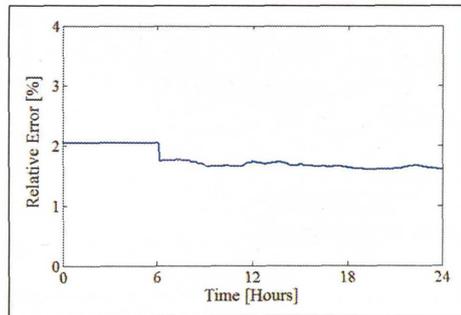


図 D.23: 2012年5月8日(火)の相対誤差

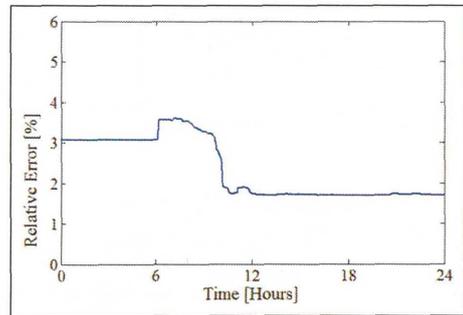


図 D.24: 2012年5月9日(水)の相対誤差

## 付録E 相対誤差と時刻の関係

2012年4月16日(月)~2012年5月9日(水)の予測データに対して逐次更新を行った時の相対誤差と時刻の関係を示す。

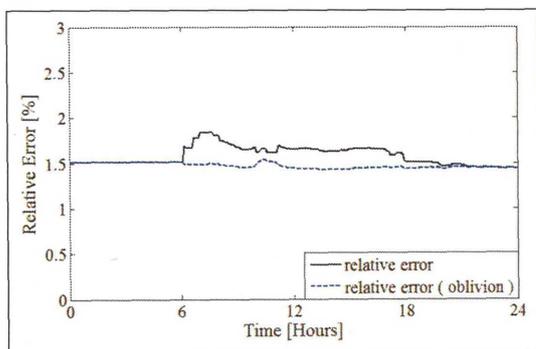


図 E.1: 2012年5月18日(金)の予測結果

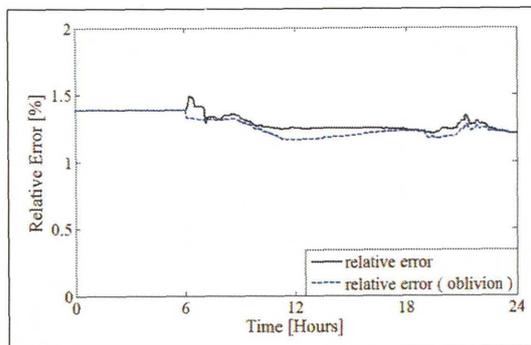


図 E.2: 2012年5月19日(土)の予測結果

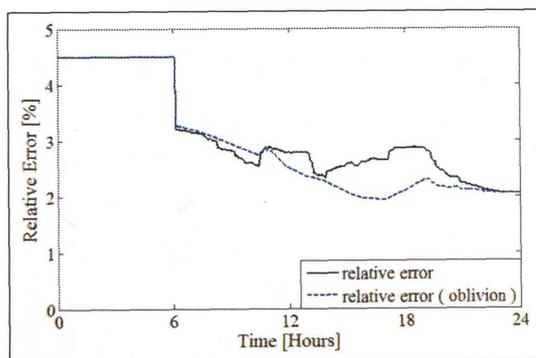


図 E.3: 2012年5月20日(日)の予測結果

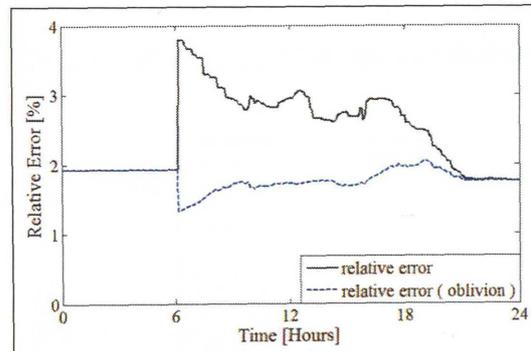


図 E.4: 2012年5月21日(月)の予測結果

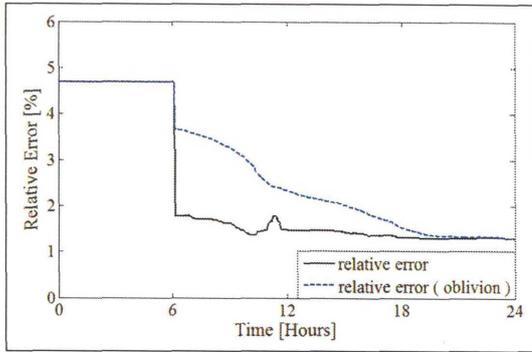


図 E.5: 2012年5月22日(火)の予測結果

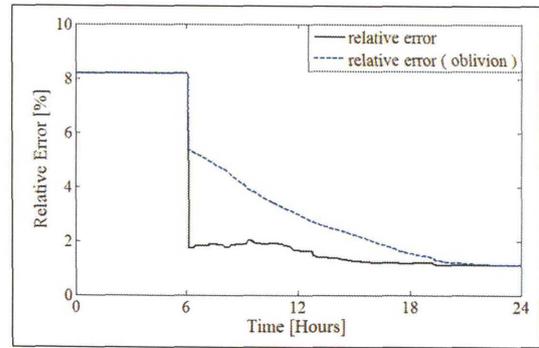


図 E.6: 2012年5月23日(水)の予測結果

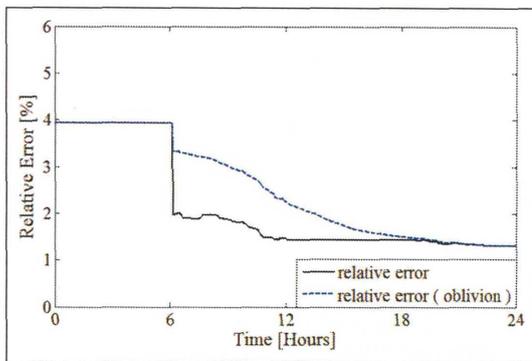


図 E.7: 2012年5月24日(木)の予測結果

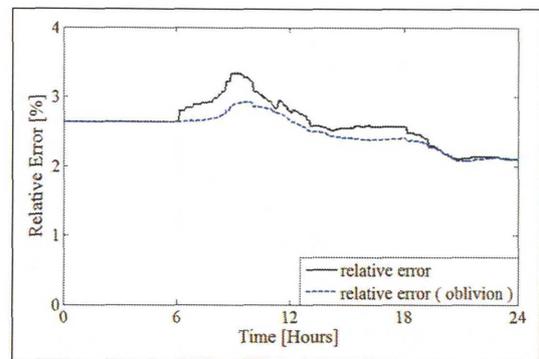


図 E.8: 2012年5月25日(金)の予測結果

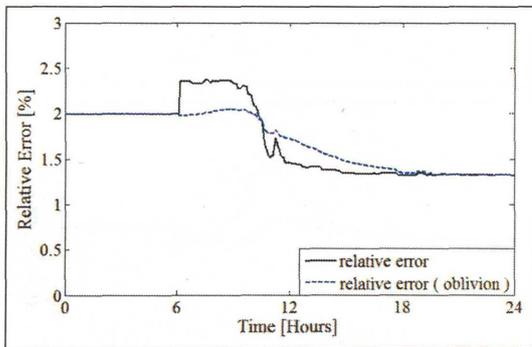


図 E.9: 2012年5月26日(土)の予測結果

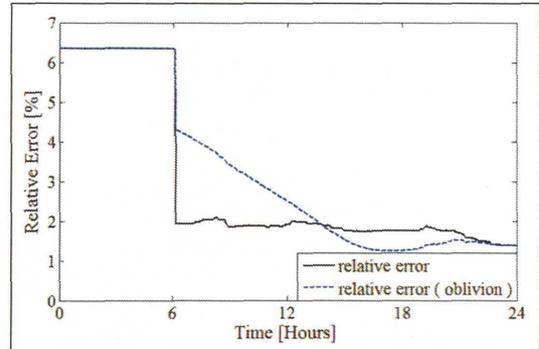


図 E.10: 2012年5月27日(日)の予測結果

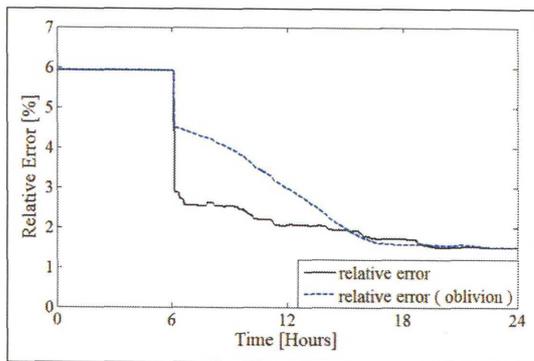


図 E.11: 2012年5月28日(月)の予測結果

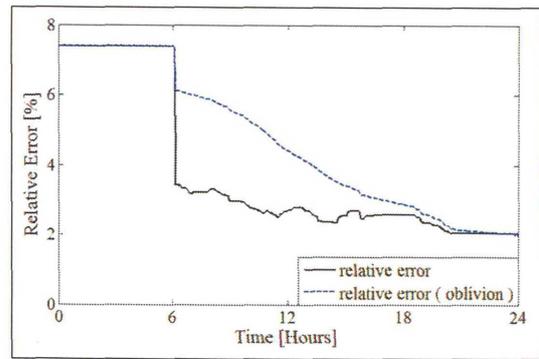


図 E.12: 2012年5月29日(火)の予測結果

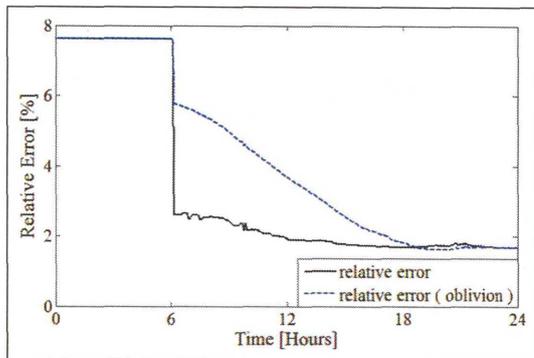


図 E.13: 2012年5月30日(水)の予測結果

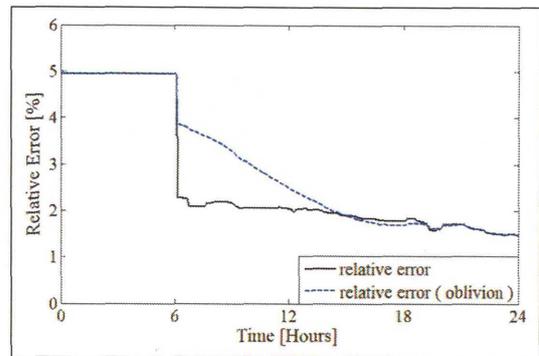


図 E.14: 2012年5月31日(木)の予測結果

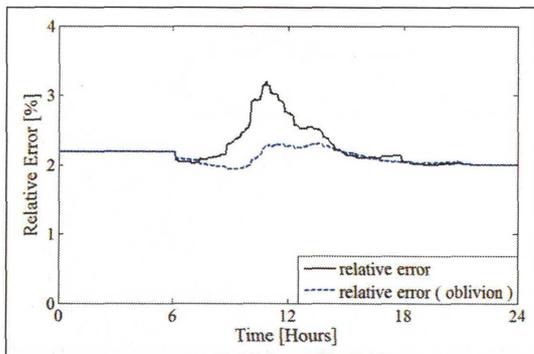


図 E.15: 2012年6月1日(金)の予測結果

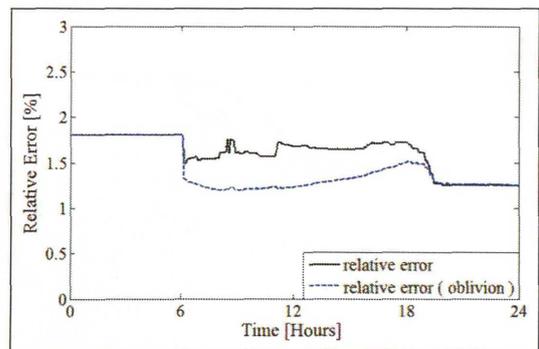


図 E.16: 2012年6月2日(土)の予測結果

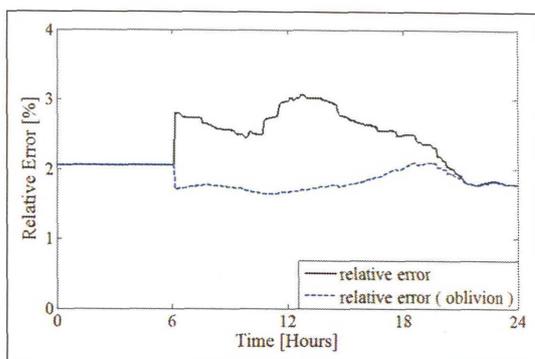


図 E.17: 2012年6月3日(日)の予測結果

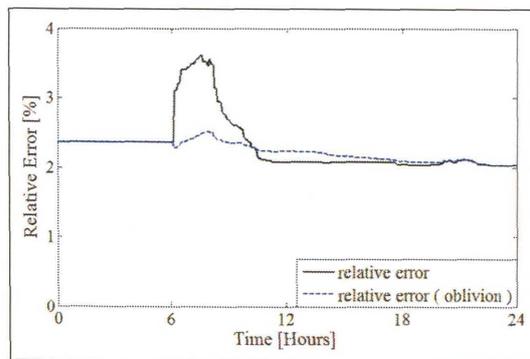


図 E.18: 2012年6月4日(月)の予測結果

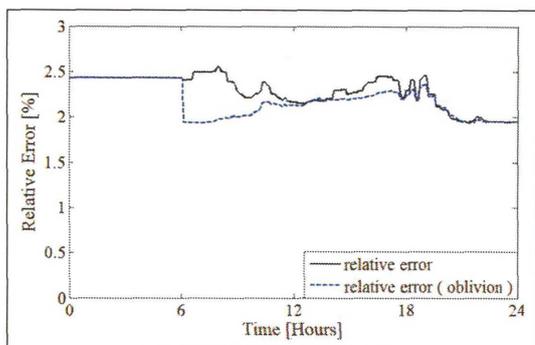


図 E.19: 2012年6月5日(火)の予測結果

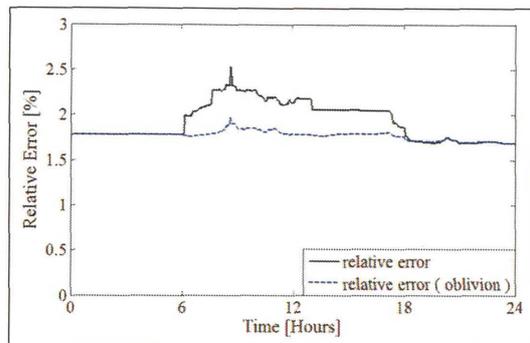


図 E.20: 2012年6月6日(水)の予測結果

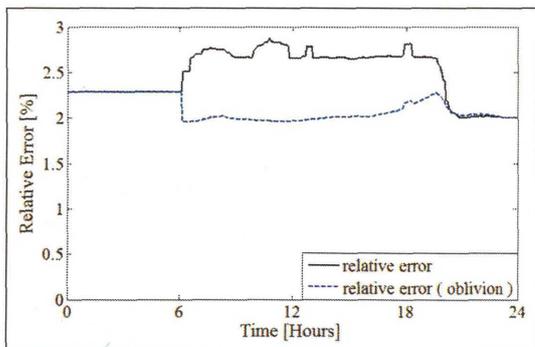


図 E.21: 2012年6月7日(木)の予測結果

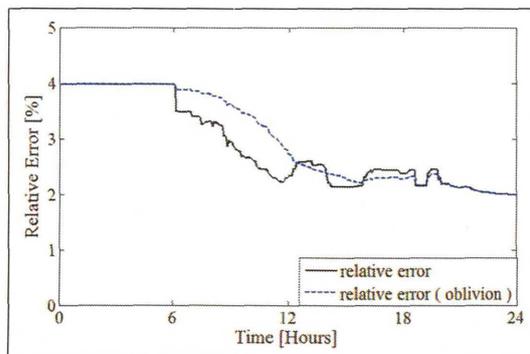


図 E.22: 2012年6月8日(金)の予測結果