

数学は工学の未来を支える知の道具

大学や大学院で工学を学ぶ学生や工学分野の技術者の皆さんにとって、数学は避けて通れない存在です。しかし、「数学は抽象的で難しい」という声を耳にすることがあります。確かに、数式は一見すると無機質で、複雑な記号の羅列に見えます。しかし、視点を変えれば、数学は工学を支える強力な道具であり、数式は万国共通で通用する言語と捉えることもできます。これらの道具や言語を活用することで、創造力を広げることができると考えられないでしょうか？

例えば、通信工学における信号処理は、フーリエ変換という数学的概念なしには語れません。複雑な時間領域の波形を周波数領域の波形に変換するこの手法は、電波の世界のみならず、音声や画像の世界まで広く応用が可能です。行列や固有値は、アンテナのビームフォーミングやネットワーク解析を語る上で不可欠です。これらは単なる計算ではなく、現象を理解し、設計を最適化する上で強い味方となる「レンズ」にも成り得るのです。

数学を味方にする第一歩は、「公式を覚える」ことから「意味を理解する」への転換だと考えます。数式は自然やシステムの構造を簡潔に表現する手段であり、その背後には必ず「なぜそうなるのか」という物語があります。例えば、微分方程式は変化の法則を記述するものです。電気回路の過渡現象や熱伝導の解析に登場する数式は、現象のダイナミズムを映し出しています。

もう一つのポイントは、数学を「便利な道具」として使う意識です。複雑な問題に直面したとき、数学は解決への近道を提供してくれます。最適化理論は、資源配分やネットワーク設計において、膨大な選択肢から最良の解を導きます。確率・統計は、通信品質の評価やAIの学習に欠かせません。これらは、現代工学の課題を解くための強力な武器なのです。

本小特集では、工学分野で数学を活用していらっしゃる専門家の解説記事をそろえました。どうか、数学を楽しむ心を忘れないで下さい。美しい定理やエレガントな証明に触れたとき、そこには芸術に似た感動があります。数学は「難しい敵」ではなく、「頼れる仲間」です。是非、数式を恐れず、その意味を味わい、工学の未来を切り開く力にして下さい。

最後に、本小特集の発行にあたり、御執筆頂いた皆様、査読や校閲に御協力頂いた皆様、編集作業に御協力頂いた皆様に感謝申し上げます。

小特集編集チーム

沢田健介、飯田大輔、飯田裕之、金友 大、
羽多野一磨、早馬道也、東野武史

円の交差グラフ: 数理とアルゴリズム

Intersection Graphs of Disks: Mathematical Properties and Algorithms

岡本吉央 Yoshio Okamoto[†]

Summary

円の交差グラフは、無線ネットワークのトポロジーを抽象化する際によく現れる。本稿では、円の交差グラフと単位円の交差グラフに関する基礎的な性質を短くまとめる。まず、ほかのグラフクラスとの関係として、平面的グラフが円の交差グラフであり、格子グラフが単位円の交差グラフであることを説明する。そして、最大独立集合問題と最小彩色問題に対して近似保証のあるアルゴリズムを説明する。最後に、最大クリーク問題が単位円の交差グラフに対しては多項式時間で解けることを説明する。最大クリーク問題が円の交差グラフに対して多項式時間で解けるかどうかは大きな未解決問題である。

Key Words

円グラフ, 単位円グラフ, 交差グラフ, 独立集合, 彩色, クリーク, 近似アルゴリズム, NP-困難性

1 はじめに

無線ネットワークにおける多くの問題では、次のように構成されるグラフを考える。その頂点は平面上の1点に対応し、無線ネットワークにおいて通信を行いたい主体を表す。そして、2頂点が辺で結ばれるのは、対応する2点間で通信が可能であるときとする。

シミュレーション等でよく用いられる単純なケースは、各点が通信可能な範囲を、その点を中心とする円で表すものである。そのときに得られるグラフは「円の交差グラフ」(intersection graph of disks)として離散数学ではよく研究されている。これは単に「円グラフ」(disk graph)と呼ばれることの方が多い。しかし、全く別の用語である pie chart に対しても同じ「円グラフ」という日本語が使われ、しかも、後者の方が広く使われているので、本稿では「円の交差グラフ」という呼び方を一貫して用いることにする。

より単純なケースとして、各円の半径が全て同一である場合を考えることも多い。拡大・縮小することで、その半径は1であると仮定して(数学的には)一般性を失わない。そのときに得られるグラフは「単位円の交差グラフ」(intersection graph of unit disks)と呼ばれる。短く「単位円グラフ」(unit disk graph)ともよく呼ばれるが、本稿では

一貫して「単位円の交差グラフ」と呼ぶことにする。離散数学において「ランダム幾何グラフ」(random geometric graph)と呼ばれる研究対象があるが、それは多くの場合、ランダムに得られた単位円の集合に対する交差グラフを意味する。

本稿の主目的は、円の交差グラフと単位円の交差グラフに対して、離散数学とアルゴリズム理論において知られていることの中で、特に基礎的なものを概観することである。次の章で円の交差グラフと単位円の交差グラフを定義した後、**3.**では、それらが成すクラスや他のグラフのクラスとの関係を考察する。続く三つの章では、最大独立集合問題、最大クリーク問題、最小彩色問題という典型的なグラフ最適化問題を考察対象とする。特に、近似アルゴリズムと近似比の考え方を導入し、定数近似比を達成するアルゴリズムの設計法を解説する。

2 円の交差グラフと単位円の交差グラフ

本稿において、「グラフ」とは二つの集合の対 (V, E) で、 E が「 V の部分集合で要素数が2のものをいくつか集めてできたものである」という性質を満たすもののことである。集合 V の要素を「頂点」と呼び、集合 E の要素を「辺」と呼ぶ。本稿では、 V は必ず有限集合であるとするが、一般に V は無限集合であっても構わない。頂点 u, v が辺 $\{u, v\} \in E$ を成すとき、 u, v は「隣接」という。

グラフを図で表すとき、頂点を平面上の点として、辺を平面上の曲線とすることが多い。このとき、辺 $\{u, v\}$ は頂点 u, v を表す点を結ぶようにしな

[†] 電気通信大学, 調布市
The University of Electro-Communications,
Chofu-shi, 182-8585 Japan

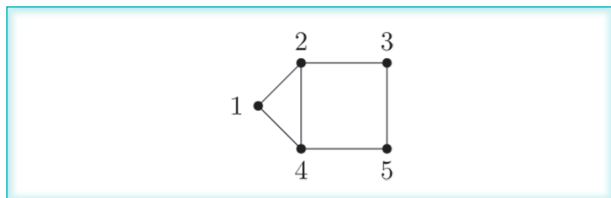


図1 グラフの例

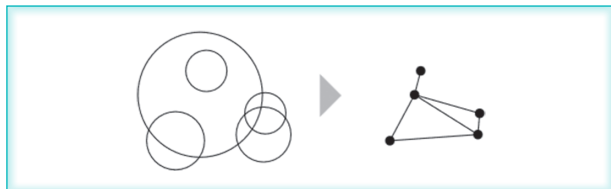


図2 円の交差グラフの例

くてはならない。辺を表す曲線同士が交わってもよい。図1が表すグラフでは1, 2, 3, 4, 5がそれぞれ頂点であり, $\{1, 2\}, \{1, 4\}, \{2, 3\}, \{2, 4\}, \{3, 5\}, \{4, 5\}$ がそれぞれ辺である。

円の交差グラフとは次のように構成されるグラフである。各頂点は平面上の円一つに対応する。また、二つの頂点が辺を成すのは、対応する二つの円が交わる（共有点を持つ）とき、そして、そのときに限る。ここで「円」と呼んでいるものは中身の詰まった閉円板を意味し、その境界である円周を指しているわけではないことに注意する。図2に例を挙げる。

単位円の交差グラフとは、各頂点に対応する円がどれも単位円であるような場合に得られる、円の交差グラフである。つまり、単位円の交差グラフは特殊な円の交差グラフである。標語的に書けば、「単位円の交差グラフ $\Rightarrow$ 円の交差グラフ $\Rightarrow$ グラフ」という関係が成り立つ。これらの逆が成り立たないことは次の章で説明する。

以下、円の交差グラフや単位円の交差グラフに対するアルゴリズムを考える際には、それらがどのようにして入力されるのかということをしっかりと考える必要がある。（その理由は後述する。）入力の方法として最も自然なものは円の集合を与えるというものである。このとき、各円はその中心の座標と半径という三つの実数値で与えられると思えばよい。一方で、少し不自然かもしれないが、交差グラフとして構成されたグラフ $G = (V, E)$ を入力として与えるという方法もある。これは座標や半径という数値を考えないため、より組合せ的な方法である。

次の意味で、円の集合を与える方が、交差グラフを与えるよりも、多くの情報をアルゴリズムに与えている。まず、円の集合が与えられれば、その中の

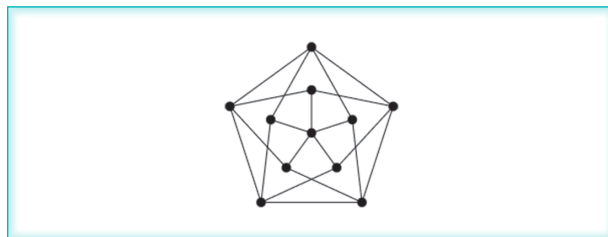


図3 Grötsch グラフ

二つの円が交わるかどうかは容易に分かるので、その交差グラフを構成することは容易にできる。ここで「容易に」というのは「多項式時間で」という意味である。一方で、交差グラフを与えても、各頂点に対応する円の中心座標や半径を復元することは難しい。ここで「難しい」というのは計算複雑性の意味での困難性を指す。例えば、与えられたグラフが円の交差グラフとして構成可能であるか、単位円の交差グラフとして構成可能であるかを問う判定問題は NP- 困難である^{(1), (2)}。NP- 困難であることの定義は省略するが、その帰結として、 $P \neq NP$ であるならば）NP- 困難である問題に対して多項式時間アルゴリズムが存在しないことが導ける。イメージで言えば、円の集合を交差グラフの設計図だと思ったとき、設計図からグラフという製品を作るのは簡単だが、製品から設計図を復元するのは難しい、ということである。

つまり、直感に訴える言い方をすれば、「円の集合を与える場合のアルゴリズムの方が、交差グラフを与える場合よりも設計しやすい」ことになる。

3 円の交差グラフの性質

円の交差グラフとして構成できないグラフはあるのだろうか？ 答えは Yes である。そのようなグラフを構成する方法はいろいろとあるが、それが本当に円の交差グラフではないことを証明しようとする、幾何学的に少し込み入った議論か、グラフ理論的な前提知識か、どちらかを必要とするので、本稿では証明しないことにする。ただ、構成の一つは紹介する。これは Grötsch グラフ（図3）と呼ばれるもので、グラフ理論においてある種の反例としてよく使われるものである。

単位円の交差グラフとして構成できないが、円の交差グラフとして構成できるグラフの存在も知られている。これについては構成の一つを証明付きで紹介する。図4にあるグラフを見てもらいたい。このグラフはよく $K_{1,6}$ と表記されるもので、一つの頂点に隣接する六つの頂点が、互いに隣接していな

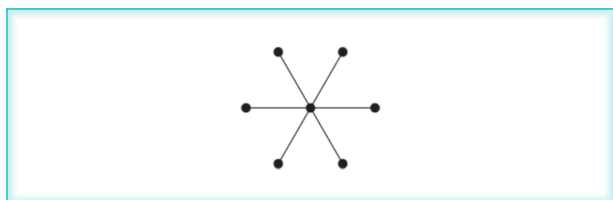


図4 単位円の交差グラフとして構成できないグラフの例 ($K_{1,6}$)

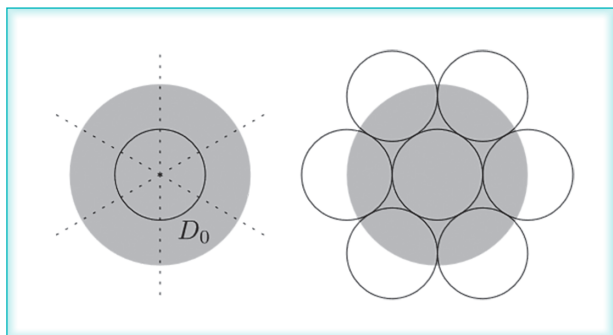


図5 $K_{1,6}$ が単位円の交差グラフとして構成できないことの説明

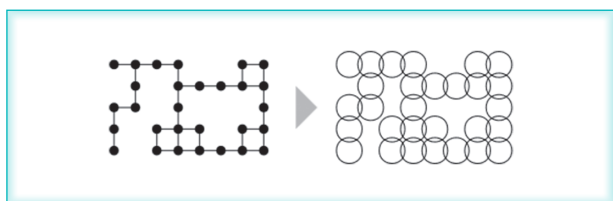


図6 格子グラフは単位円の交差グラフである

い。

これが単位円の交差グラフとして構成できると仮定しよう。中央の頂点に対応する単位円を D_0 とする。ほかの六つの頂点に対応する単位円は D_0 と交差しないといけないので、それらの中心は D_0 の中心を中心とする半径2の円の中（または境界上）になければならない。その円を頂角 60° の扇形で六等分すると、一つの扇形に中心を持つ二つの単位円は必ず交差することが分かる（図5左）。そのため、 D_0 に交差する単位円で互いに隣接しないものはそれぞれ別の扇形に含まなければならない。しかし、図5右に示すようにそれは不可能である。したがって、 $K_{1,6}$ は単位円の交差グラフとして構成できない。

格子グラフは単位円の交差グラフとして実現できる。ここで、格子グラフとは次のように構成されるグラフのことである。その頂点集合は二次元整数格子 $\mathbb{Z}^2$ の有限部分集合であり、二つの頂点が辺を成すのは、その距離が1であるとき、そして、そのときに限る。図6は格子グラフの例であるが、それを単位円の交差グラフとして実現した様子も描い

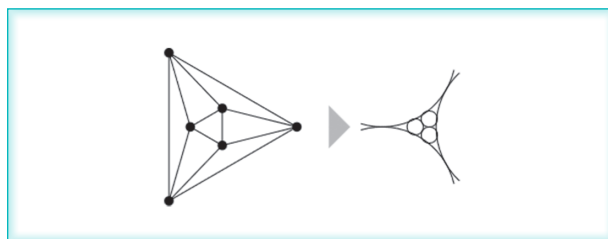


図7 平面的グラフは円の交差グラフとして実現できる

ている。つまり、標語的に書けば「格子グラフ $\Rightarrow$ 単位円の交差グラフ」という関係の成立が分かる。一方で、その逆は一般に成り立たず、単位円の交差グラフであるが、格子グラフとして実現できないものは簡単に構成できる。

平面的グラフは円の交差グラフとして実現できる。ここで、平面的グラフとは、図として表すときに、辺を表す曲線同士が交わらないようにできるもののことである。つまり、標語的に書けば「平面的グラフ $\Rightarrow$ 円の交差グラフ」という関係が成立するわけである。この事実は「Koebe-Andreiev-Thurston の定理⁽³⁾」と呼ばれるものの帰結であるが、その定理の証明は本稿の目的を逸脱する（そのため、行わない）。例を図7に示す。

ここまで述べてきたことを鑑みると、次の帰結が得られる。

- ・格子グラフに対して解くのが難しい問題は、単位円の交差グラフに対しても解くのが難しい。ただし、単位円の交差グラフは「円の集合」として与えられるものとしても、「グラフ」として与えられるものとしてもよい。
- ・平面的グラフに対して解くのが難しい問題は、円の交差グラフに対しても解くのが難しい。ただし、円の交差グラフは「グラフ」として与えられるものとする。

4 最大独立集合問題

グラフ $G = (V, E)$ の「独立集合」とは、頂点の集合 $S \subseteq V$ で、その中のどの2頂点も辺を成していないもののことである。つまり、任意の $u, v \in S$ に対して、 $\{u, v\} \notin E$ が成り立つことである。グラフ G の独立集合の中で、要素数が最大のものを G の「最大独立集合」と呼ぶ。グラフ G の最大独立集合の要素数を $\alpha(G)$ で表すことが多く、これを G の「安定数」と呼ぶことがある。

最大独立集合問題とは、入力として与えられたグラフ G に対して、 G の最大独立集合を出力することを要求する問題である。

最大独立集合問題は、単位円の交差グラフに対して、それが「円の集合」として与えられる場合であっても NP- 困難であることが知られている⁽⁴⁾。つまり、多項式時間で厳密に最大独立集合を出力するアルゴリズムを設計することは困難であると思われる。そのため、厳密に最大独立集合を出力するのではなく、最大独立集合に近い独立集合を出力するアルゴリズムを設計する試みが多くなされている。

この「近さ」を定量化するために、近似比という概念がよく使われている。あるアルゴリズム A が、入力されたグラフ G の独立集合 S を出力するとする。このとき、A が近似比 c を達成するとは、任意の G に対して $\alpha(G) \leq c \cdot |S|$ が成り立つことである。ここで、 $|S|$ は独立集合 S の要素数を表す。最大独立集合の要素数を $\alpha(G)$ と定義したので、 $\alpha(G) \geq |S|$ が必ず成り立つことに注意する。つまり、近似比 c は必ず 1 以上の実数であり、それが 1 に近ければ近いほど、近似の精度が良いと言える。近似アルゴリズムを設計する際に重要となるのは、計算量が小さいことに加えて、近似比が小さいことである。

単位円の交差グラフがグラフによって与えられるとき、次の貪欲アルゴリズム 1 を考える。

〈アルゴリズム 1〉

- (1) $S = \emptyset$ とする。
- (2) 頂点 $v \in V$ を一つ任意に選び、 S に追加する。
- (3) v と v に隣接する全ての頂点を G から（すなわち V から）削除する。
- (4) $V = \emptyset$ になったら、 S を出力して終了する。
そうでなければ、ステップ 2 に戻る。

アルゴリズム 1 は近似比 5 を達成する⁽⁵⁾。その理由を説明する。与えられた単位円の交差グラフ G の最大独立集合の一つを S^* とし、アルゴリズム 1 の出力を S とする。アルゴリズム 1 において、 S に頂点 v が追加されたとき（つまり、ステップ 2）を考える。このとき、次のステップ 3 で v と v に隣接する全ての頂点を G から削除するが、その中には、 S^* の頂点が高々 5 個しか存在しない。これは次のように分かる。頂点 v が S^* の要素であるとき、 v に隣接する頂点はどれも S^* の要素ではないので（要素であるとするとき S^* が独立集合であることに矛盾する）、削除した頂点の中で S^* の頂点であるのは一つである。一方、 v が S^* の要素でないとき、 v の隣接頂点の中に S^* の頂点が 6 個あると、それらと v が $K_{1,6}$ を作ってしまい、これは単位円の交差グラフではないので、矛盾が導いてしまうのである。したが

って、 S の頂点一つに S^* の頂点が高々五つ対応するので、 $5|S| \geq |S^*| = \alpha(G)$ の成立が分かる。

アルゴリズム 1 の達成する近似比を導出する際には、「任意の頂点 v に対して、 v に隣接する頂点の中には S^* の頂点が高々五つしか存在しない」という性質が重要であった。実を言うと、単位円の交差グラフにおいては「ある頂点 v に対して、 v に隣接する頂点の中には S^* の頂点が高々三つしか存在しない」ことが成り立つ。（単位円の集合の中で、中心の x 座標が最小のものを考えればよい。）そのような頂点は多項式時間で見つけれられるので、アルゴリズム 1 のステップ 1 において「任意に選ぶ」ところを上で存在を保証した頂点を選ぶように変えると、同じ導出によって、近似比 3 を達成できることが分かる⁽⁵⁾。

単位円であるとは限らない円の交差グラフに対しては、ステップ 2 で頂点 v を「半径最小の円に対応する頂点」とすることで、近似比 5 を達成できる。証明は単位円の場合と同様である。しかし、そのような円を見つけるためには、交差グラフが「円の集合」として与えられる必要がある。一方で、円の集合として与えられなくても、半径最小の円に対応する頂点は「その隣接頂点に制限すると、頂点数が 6 以上の独立集合を持たない」という性質を持ち、そのような頂点を見つけることは多項式時間で行うことができる。このとき、ステップ 2 における v をそのような頂点とすると、同じ証明によって、近似比 5 を達成できることが分かる⁽⁵⁾。

実を言うと、もっと小さな近似を多項式時間で達成するアルゴリズムが知られている。それは究極とも言えるもので、つまり、任意の実定数 $\epsilon > 0$ に対して、近似比 $1 + \epsilon$ を達成する多項式時間アルゴリズムである。そのようなアルゴリズムは「多項式時間近似スキーム」(PTAS: Polynomial-Time Approximation Scheme, ピータス) と呼ばれることもある。ただし、「多項式時間」の多項式の次数は ϵ に依存してもよいので、その点には注意が必要である。典型的には、 ϵ が小さいほど多項式時間近似スキームの計算量は大きくなる。

円の交差グラフにおける最大独立集合問題に対して、円の集合を入力とするとき多項式時間近似スキームの存在が知られている⁽⁶⁾。アルゴリズムは少し複雑であるため、紹介しない。単位円の交差グラフの場合は、グラフを入力とするときでさえ多項式時間近似スキームが存在する⁽⁷⁾。円の交差グラフにおいて、グラフを入力とするときに多項式時間近似スキームが存在するのか、筆者は知らない。

5 最大クリーク問題

グラフ $G = (V, E)$ の「クリーク」とは、頂点の集合 $S \subseteq V$ で、その中のどの 2 頂点も辺を成しているもののことである。つまり、任意の $u, v \in S$ に対して、 $\{u, v\} \in E$ が成り立つことである。グラフ G のクリークの中で、要素数が最大のものを G の「最大クリーク」と呼ぶ。グラフ G の最大クリークの要素数を $\omega(G)$ で表すことが多く、これを「 G のクリーク数」と呼ぶことがある。

最大クリーク問題とは、入力として与えられたグラフ G に対して、 G の最大クリークを出力することを要求する問題である。

最大クリーク問題は、単位円の交差グラフに対して、それがグラフとして与えられる場合であっても、多項式時間で厳密に解けることが知られている⁽⁴⁾。一方で、円の交差グラフにおいて最大クリーク問題が（円の集合として与えられる場合でも、グラフとして与えられる場合でも、）多項式時間で解けるか、NP- 困難であるか、ということは大きな未解決問題である。

単位円の交差グラフにおいて最大クリーク問題を厳密に解くアルゴリズムを短く紹介する。アルゴリズムには単位円の交差グラフがグラフ $G = (V, E)$ として与えられる。まず、グラフの各辺 $\{u, v\} \in E$ を考えて「 u, v を含むクリークの中で最大のもの」を見つける手順を考える。そして、それを全ての辺に対して行い、その中で最大のものを出力すれば、グラフ G の最大クリークが見つけれられるのである。各辺に対する手順が多項式時間で実行できれば、辺の数は多くても二つの頂点を選ぶ組合せの総数しかないので、アルゴリズム全体の計算量も多項式となる。

ここで、辺 $\{u, v\} \in E$ を任意に一つ固定する。頂点 u, v に対応する単位円を D_u, D_v とすると、この二つの円は交わっていることになる。頂点 u, v を含むクリークに含まれる頂点に対応する単位円は D_u, D_v の両方と交わらないとならないため、そのような単位円の中心は図 8 の灰色で示される領域にしか存在できない。

灰色の領域を図 8 のように二等分すると、上の領域に中心を持つ単位円は互いに交差し、下の領域に中心を持つ単位円も互いに交差することが分かる。（各領域における 2 点の距離の最大値が 2 以下であるため。）したがって、 u, v の両方と隣接する頂点全体に制限すると、グラフは「二部グラフの補グラフである」という性質を持つ。二部グラフや補グラフの定義や説明は省くが、重要なことは、「二

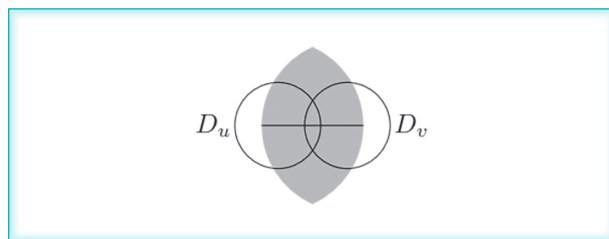


図8 最大クリーク問題を解くアルゴリズムの説明

部グラフの補グラフにおいて、最大クリーク問題は多項式時間で解ける」ということである。実際、二部グラフの補グラフにおける最大クリーク問題は、二部グラフの最大マッチング問題に帰着できるのだが、二部グラフの最大マッチング問題は増加道法というアルゴリズムによって多項式時間で解けることが知られているのである。（増加道法の説明はグラフ理論の教科書に大抵載っている。）これによって、単位円の交差グラフにおいて最大クリーク問題が多項式時間で解けることが分かった。

以下、増加道法を知る人に対して、その帰着を説明する。二部グラフの最大マッチング問題に対する増加道法を実行すると、その出力として、最大マッチングだけでなく、その双対に対応する最小頂点被覆も得られる。最小頂点被覆の補集合は最大独立集合であるので、最大独立集合も得られる。二部グラフの最大独立集合はその補グラフの最大クリークである。つまり、二部グラフの補グラフの最大クリークが増加道法から得られるのである。

円の交差グラフに対しては、最大クリーク問題が多項式時間で解けるかどうか分かっていない。最近の研究によって、乱択多項式時間近似スキームが存在すること⁽⁸⁾ や、異なる半径の種類が定数であるときには多項式時間で解けること⁽⁹⁾ が分かっている。

6 最小彩色問題

グラフ $G = (V, E)$ の「彩色（さいしょく）」とは、頂点への色の割当 $f: V \rightarrow C$ で（ただし、 C は色の集合）、任意の辺 $\{u, v\} \in E$ に対して、 $f(u) \neq f(v)$ を満たすもの、つまり、辺の両端点の色が必ず異なるもののことである。使う色の数、つまり、像 $f(V)$ の要素数が最小である彩色 f を G の「最小彩色」と呼ぶ。最小彩色が使う色数を $\chi(G)$ で表し、「 G の染色数」と呼ぶことがある。彩色において、同じ色が割り当てられている頂点の集合は独立集合であることに注意する。

最小彩色問題とは、入力として与えられたグラフ G に対して、 G の最小彩色を出力することを要求す

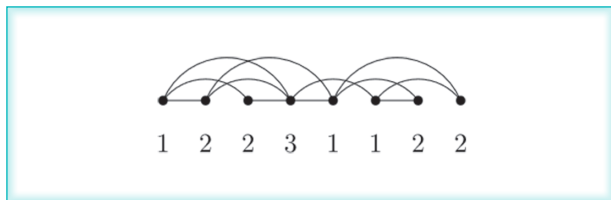


図9 アルゴリズム2の実例 左から右に向かって頂点が並んでいると見る。

る問題である。

最小彩色問題は、単位円の交差グラフに対して、それが単位円の集合として与えられる場合であってもNP-困難であることが知られている⁽⁴⁾。そのため、多項式時間近似アルゴリズムが研究されている。ここで、グラフの彩色を出力するアルゴリズムAが近似比 c を達成するとは、任意のグラフ G に対して、使う色数が $c \cdot \chi(G)$ 以下の彩色を出力することである。

まず、交差グラフとは限らず、グラフ一般に対して、彩色を得る手順をアルゴリズム2として説明する。このアルゴリズム2は最小彩色を出力するわけではないことに注意する。ここで、記述を簡単にするため、色の集合は正整数全体の集合、つまり、 $C = \{1, 2, 3, \dots\}$ であるとする。

〈アルゴリズム2〉

- (1) 頂点を一列に並べて、 $v_1, v_2, \dots, v_n$ とする。
- (2) 全ての $i = 1, 2, \dots, n$ について、順に $f(v_i) = \min(C \setminus \{f(v_j) \mid j < i, \{v_j, v_i\} \in E\})$ とする。
- (3) 彩色 f を出力する。

ステップ2で行っていることを言葉で述べると、頂点 v_i の色を決めるとき、 v_i の隣接頂点で既に使われた色以外で最も（数値として）小さい色を v_i の色、つまり、 $f(v_i)$ とするのである。アルゴリズム2で彩色を行った例を図9に示す。

アルゴリズム2が彩色を必ず出力することは次のように分かる。辺 $\{v_j, v_i\} \in E$ に着目する。（ただし、 $j < i$ とする。）頂点 v_i の色 $f(v_i)$ を決めるとき、 v_j の色 $f(v_j)$ は既に決まっている。ステップ2で $f(v_i)$ には $f(v_j)$ と異なる色が割り当てられるので、 $f(v_i) \neq f(v_j)$ となる。つまり、アルゴリズム2の出力 f は彩色である。

アルゴリズム2では、ステップ1で頂点を一列に並べる。この並べ方によって出力する彩色が使う色数が変わる。実を言うと、ある並べ方に対してアルゴリズム2を実行すると最小彩色が得られる。つまり、最小彩色問題の難しさは、そのような並べ

方を見つけることの難しさと同じなのである。

アルゴリズム2が出力する彩色 f が使う色数を評価するために、グラフにおける頂点の次数を定義する。グラフ $G = (V, E)$ の頂点 $v \in V$ の次数とは、 v に隣接する頂点の総数であり、ここでは $d(v)$ で表す。グラフ G の最大次数とは $\max\{d(v) \mid v \in V\}$ のことであり、ここでは $\Delta(G)$ で表す。例えば、図9のグラフにおいて、最大次数は5である。

アルゴリズム2の出力が使う色数は必ず $\Delta(G) + 1$ 以下であることが次のように分かる。頂点 v_i の色を決めるとき、使えない色数は高々 $d(v_i)$ である。したがって、 $f(v_i) \leq d(v_i) + 1 \leq \Delta(G) + 1$ が成り立つ。

アルゴリズム2を G が単位円の交差グラフであるときに適用すると、出力する彩色の使う色数が $5\chi(G) - 4$ 以下になることを証明する⁽⁵⁾。すなわち、アルゴリズム2は単位円の交差グラフに対して近似比5を達成する。

証明のための補題として、 G が単位円の交差グラフであるとき、 $\chi(G) \geq \Delta(G)/5 + 1$ が成立することを示す。これが証明できると、先に説明したとおり、アルゴリズム2の使う色数が高々 $\Delta(G) + 1$ であることから、それが高々 $\Delta(G) + 1 \leq 5\chi(G) - 4$ であることが導ける。

では、補題を証明する。単位円の交差グラフ G において、 f^* を最小彩色として、 v^* を次数が最大の頂点（つまり、 $d(v^*) = \Delta(G)$ ）とする。頂点 v^* に対応する単位円を D^* とすると、 D^* と交わる単位円の色が $\Delta(G)$ である。彩色において、同じ色が割り当てられる頂点の集合は独立集合であるので、それら $\Delta(G)$ 個の頂点の中で同じ色を持つ頂点の数は高々5である。（そうでないと、 $K_{1,6}$ を単位円の交差グラフとして構成できてしまう。）つまり、 v^* に使う色と合わせて、 $\chi(G) \geq \Delta(G)/5 + 1$ が得られる。

並べ方を工夫することで、近似比を下げるができる。頂点を並べるとき、対応する単位円の中心の x 座標が小さい方から順に並べるとする。こうすると、近似比3を達成できる。ただ、この並べ方を実現するためには、単位円の交差グラフが単位円の集合として与えられる必要がある。

近似比3を達成できる理由を説明する⁽⁵⁾。頂点 v_i の色を決めるとき、それよりも中心が左にあるものには既に色が割り当てられていることになるが、 v_i に対応する単位円と交わり、中心がそれより左にあるものは頂角が 60° の三つの扇形の中に中心を持つことになる。一つの扇形に中心を持つ単位円同士は交わるので、そこには高々 $\omega(G) - 1$ 個の単位円

しか存在しないことになる。($\omega(G)$ は G のクリーク数である。) したがって、 v_i に隣接し、既に色が割り当てられている頂点の数は高々 $3\omega(G) - 3$ であり、アルゴリズム 2 が v_i に割り当てる色は $3\omega(G) - 2$ 以下である。どんな彩色においてもクリーク数だけは色を使う必要があるので、 $\omega(G) \leq \chi(G)$ が成り立ち、したがって、アルゴリズム 2 が使う色数は高々 $3\chi(G) - 2$ であることが導ける。

今、この証明から、 G が単位円の交差グラフであるとき、 $\chi(G) \leq 3\omega(G) - 2$ が成立することが分かる。このように、 $\chi(G)$ を $\omega(G)$ の関数として上から抑えることは様々なグラフクラスに対して研究対象となっている。単位円の交差グラフに対してはその関数が $3\omega(G) - 2$ であるわけだが、ここに現れる $\omega(G)$ の前の係数を 3 より小さくできるかということとはよく分かっていない。大きな未解決問題であると認識されている。

単位円であるとは限らない円の交差グラフを考える。頂点の並べ方として「半径の大きな円から順に並べる」というものを考えると、単位円の交差グラフのときと同じ考え方で、近似比 5 を達成できることが分かる⁽⁵⁾。この並べ方でアルゴリズム 2 を実行するためには、円の集合として交差グラフが与えられる必要がある。筆者の知る限り、5 が多項式時間で今のところ達成できる最も良い近似比である。

7 最後に

単位円の交差グラフと円の交差グラフについて、基礎的な事項を概観した。その中で、このように簡単な幾何学的対象についても未解決問題が幾つかあることも紹介した。

本稿では扱わなかったが、円の集合と点の集合が与えられて、それらに対して最適化問題を解くという設定もよく考えられている。例えば、与えられた点を全て覆うように円を選ぶとき、選ばれた円の総数を最小化する、といった問題である。(「円による点の最小被覆問題」と呼ばれる。) 多くの問題は NP- 困難であるため、近似アルゴリズムがよく研究されている。

現実問題の抽象化として、円や単位円を用いることは過度な理想化であると思われるかもしれない。実際そうであると思う。しかし、そうであっても、円や単位円を使って得られた成果を足掛かりにし

て、もっと複雑な状況に対する考察を行うことができるようになる。そのためには、円や単位円という単純な場合に、強固な数学的知見を得ることが重要であり、そのために離散数学が役立つことを強調したい。

謝辞 通信ソサイエティマガジン編集委員会の方々には本稿の執筆を御推薦頂いた。査読委員、校閲委員の方には改善・修正案を頂いた。ここに感謝する。

文献

- (1) H. Breu and D. Kirkpatrick, "Unit disk graph recognition is NP-hard," *Comput. Geometry*, vol. 9, no. 1-2, pp. 3-24, Jan. 1998.
- (2) P. Hlineny and J. Kratochvil, "Representing graphs by disks and balls," *Discrete Math.*, vol. 229, no. 1-3, pp. 101-124, Feb. 2001.
- (3) P. Koebe, "Kontaktprobleme der konformen abbildung," *Ber. Verh. Sächs. Akad. Leipzig*, vol. 88, pp. 141-164, 1936.
- (4) B. Clark, C. Colbourn, and D. Johnson, "Unit disk graphs," *Discrete Math.*, vol. 86, pp. 165-177, 1990.
- (5) M. Marathe, H. Breu, H. Hunt III, S. Ravi, and D. Rosenkrantz, "Simple heuristics for unit disk graphs," *Networks*, vol. 25, pp. 59-68, 1995.
- (6) T. Erlebach, K. Jansen, and E. Seidel, "Polynomial-time approximation schemes for geometric intersection graphs," *SIAM J. Comput.*, vol. 34, no. 6, pp. 1302-1323, 2005.
- (7) T. Nieberg, J. Hurink, and W. Kern, "Approximation schemes for wireless networks," *ACM Trans. Algorithms*, vol. 4, no. 4, pp. 49:1-49:17, Aug. 2008.
- (8) M. Bonamy, E. Bonnet, N. Bousquet, P. Charbit, P. Giannopoulos, E. Kim, P. Rzazewski, F. Sikora, and S. Thomasse, "EPTAS and subexponential algorithm for maximum clique on disk and unit ball graphs," *J. ACM*, vol. 68, no. 2, pp. 9:1-9:38, Jan. 2021.
- (9) J. Keil and D. Mondal, "The maximum clique problem in a disk graph made easy," *SoCG 2025*, pp. 63:1-63:16, June 2025.

(2025 年 8 月 22 日受付, 11 月 5 日再受付)

岡本吉央 (正員)

2005 スイス連邦工科大チューリッヒ校了。Ph.D. 同年豊橋技科大工・情報工学系助手。2007 同助教。2007 東工大大学院情報理工学研究科特任准教授。2010 北陸先端大大学院教育イニシアティブセンター特任准教授。2012 電通大大学院情報理工学研究科情報・通信工学専攻准教授。2016 同情報・ネットワーク工学専攻准教授。2017 同教授。現在に至る



人の内面状態の推定と拡散モデルの応用 ——自己と他者を理解し支援することを目指して——

Estimation of Human Internal States and Application of Diffusion Models

——Toward Understanding and Supporting the Self and Others——

幸島匡宏 Masahiro Kohjima[†] 南部優太 Yuta Nambu[†]

Summary

人の感情やストレス・健康状態といった「内面状態」は、行動・意思決定・生活の質に深く関係しており、ヘルスケアや教育、マーケティングなどの様々な分野で内面状態を推定する技術の発展が期待されている。本稿では、機械学習を用いて人の内面状態を推定する技術に関心を持つ読者を対象に、基本的な問題設定及び代表的なデータと応用事例について概観するとともに、画像生成で注目を集める拡散モデルを用いたアプローチに焦点を当てる。拡散モデルの基礎的な原理を平易に解説した上で、それが内面状態推定の文脈でどのように応用されているかについて、筆者らが取り組む生体信号のデータ生成への適用事例とともに紹介する。

Key Words

内面状態、機械学習、深層生成モデル、拡散モデル

1 まえがき

人の感情やストレス、心や体の健康状態といった「内面状態」は、人の行動や意思決定・生活の質に深く関わりとされている。表情や声、チャット、行動パターン、心拍数の変化などから推定した内面状態を用いることで、安全の確保（運転者の眠気を検出）やメンタルヘルスの支援（従業員のストレスや抑うつ傾向を検出）、教育効果向上（学生の授業中の集中度や理解度を把握）やマーケティング（顧客の広告や商品、イベントへの関心度を把握）など様々な場面で、人に寄り添い支援する手段として活用することが期待されている^{(1), (2)}。

内面状態の推定に関する研究は学際的な領域であり、心理学・人工知能・生体センシング・HCIなどの分野と深く関連している。本稿では筆者の専門分野である機械学習（人工知能の一分野）の観点に注目する。

機械学習の観点から見ると、内面状態推定は、人が表現した何らかの情報（画像、音声、テキスト、行動ログ、生体信号など）から内面状態を表すラベルを推定する分類問題、あるいは回帰問題として定式化できる。ただし、本質的に対象が人間であるということに起因して、内面状態を推定するモデルを学習するためのデータの収集は困難であることが多

い。例えば、計測対象とする人のプライバシー配慮及び負担軽減のために収集期間が制限されることや、センシングのための専用機器を個別に用意する金銭的コストの問題で、対象者の数が制限される場合がある。

上記の困難さに対処するための有望なアプローチとして、深層生成モデル、特に高品質でリアルな画像を多様に生成できることで大きな注目を集める拡散モデル^{(3), (4)}及びフローモデル^{(5)~(7)}を用いるものがある。具体的には、拡散モデルの代表例である Denoising Diffusion Probabilistic Models (DDPM)⁽³⁾とその発展手法^{(8)~(13)}や、DDPMの代替として Stable Diffusion 3^{*1}に採用された Rectified Flow (RF)^{(6), (7), (14), (15)}が挙げられる。DDPM・RFを用いることで、これ以前に存在した深層生成モデルと比較して、高品質かつ多様性に富んだデータの生成が可能となることから、内面状態を推定するモデルの学習に必要なデータの生成（データ拡張）や、人の負担が少なく、かつ安価な機器で計測された情報の別の情報への変換（モダリティ変換）に応用する試みが活発化しつつある。

本稿では、内面状態の推定及び拡散モデルを用いた応用事例について焦点を当てた解説を行う。まず **2.**で内面状態推定の基本的な問題設定と課題について述べる。**3.**からは内面状態推定の課題解決に有望なアプローチである DDPM⁽³⁾と RF⁽⁶⁾を用い

[†] NTT 株式会社人間情報研究所、横須賀市
Human Informatics Laboratories, NTT Inc.,
Yokosuka-shi, 239-0847 Japan

* 1 <https://stability.ai/news/stable-diffusion-3-research-paper>

表 1 内面推定技術に用いる計測情報の例

データ	主な特徴	具体例	各具体例の利点や欠点
画像／映像	表情や声量などの内面状態に直結する情報が得られるがプライバシー懸念が大きい	表情	FACS 理論など心理学の知見も豊富（FACS は画像認識でも活用） ^{(18), (19)}
		姿勢	会議に退屈しているなど表情には出ない無意識な反応を捉えられる ⁽²⁰⁾
音声		会話	会話内容に加え、発話の交代回数や沈黙時間なども特徴量として利用できる
		非言語発話	声帯振動等の一部病態（パーキンソン病や ALS 等）と相関指標が計算可能
テキスト	価値観など書き手の深い内面が反映される	SNS 投稿	瞬間的な感情が反映されたものなど不特定多数の人の大量の投稿が集まる
		ブログ・日記	長期的な内面状態の変化を追える可能性があるが記述されない日もある
行動ログ	センサを用いない計測が可能（自己評価も可）	PC ／スマホログ	タイピング数や起動中のアプリなどリアルタイムな計測も可能 ⁽²¹⁾
		アンケート	外出有無など自由に回答項目を設定可能（内面状態ラベルも一緒に収集可）
生体信号	表情や声に現れない微細な変化を捉えられる	心電位（ECG）	知見が豊富で様々な特徴量も算出可能だがベルト型デバイス等の装着が必要
		脈波（PPG）	腕時計型デバイス等で容易に常時計測可能だが心電位と比較して知見が不足

たデータ生成について説明し、これらを生体信号のモダリティ変換に適用した応用例^{(16), (17)}について紹介する。最後に 4. で今後の展望について述べる。

2 内面状態推定の基礎

まず初めに、内面状態推定において利用されるデータ及び基本的な問題設定について説明する。

2.1 データの作成

機械学習による内面状態推定を行うためには、まず“内面状態に関するラベル付きのデータ”を準備する必要がある。これは、人間が表現した何らかの情報（画像、音声、テキスト、行動ログ、生体信号など）を表す計測情報 x と、感情やストレス、健康状態などの内面状態を表すラベル y を対応付けたペアの集合 $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$ として表現される。ただし、 (x_i, y_i) は i 番目のペア、 n はペアの総数を表す。

計測情報：上記の計測情報 x の代表例を表 1^{(18)~(21)} に示す。内面状態との関連が強いと考えられる表情や声を計測できる画像・音声のみでなく、PC やスマートフォンを通して比較的手軽に計測できるテキストや行動ログ、表情や声に現れない微細な変化を捉えられる生体信号など、様々なものが利用される。推定対象とする内面状態や計測対象者の環境、収集期間などに応じて、これらの情報の一つまたは複数（後述する MERR⁽²²⁾ のように映像・音声・テキストの組合せなど）が利用される。

内面状態ラベル：内面状態に関するラベル y が感情に関わる場合には、心理学者 P. Ekman の基本 6 感情（喜び、悲しみ、怒り、驚き、恐れ、嫌悪）⁽²³⁾ を基に一部の感情を抜き出す／粒度を細かくして作成した離散値ラベル⁽²⁴⁾ や、Valence（快－不快）×Arousal（興奮－沈静）⁽²⁵⁾ のそれぞれの強度を表す二次元空間上の連続値のラベルが利用される⁽²⁶⁾。後者は感情の微妙な違いを表現するのに適しており、複雑な感情状態を扱う研究において広く採用されている。感情以外の、例えばストレスや健康状態に関するラベルを用いる場合、低ストレス／軽症・高ストレス／重症などの強度別の離散値、または連続値が用いられることが多い。

アノテーション：データの計測情報に内面状態ラベルを付与すること（「アノテーション」と呼ぶ。）により作成される。表 2 に代表的なデータの作成方法について示す。アノテーションはラベルを付与する主体者が誰であるかにより、大きく以下の三つに分類される。

- ・自己評価 当事者が自らの内面状態を回答する方法であり、微細な感情変化など外からの判定が困難なラベルの取得に用いられる。表 2 の WESAD⁽²⁷⁾ など映像を提示される環境下で行う場合もある。DeepMood⁽²⁸⁾ や StudentLife⁽²⁹⁾ のように長期間にわたる報告を行う場合には、「経験サンプリング法」と呼ばれる心理学・社会科学分野の手法が用いられる。この方法では、記憶の曖昧さの影響を小さくするため、ランダムまたは事前に決めたタイミン

表2 内面推定に関する研究で用いられるデータの例及び初期分析で利用された手法

データ / 手法名	規模	計測データおよび収集法 / 作成法	ラベル及び アノテーション	推定手法
Affectnet ⁽²⁶⁾	45 万画像	感情語（多言語）を用いた検索で収集した顔画像	複数熟練者による離散／連続感情	CNN 等
Parkinsons ⁽³⁰⁾	5,875 録音	被験者の持続発声” ahhh・・・”を自宅から遠隔収録	医師の観察による重症度評価	線形回帰, CART
GoEmotions ⁽²⁴⁾	58,000 件	米国の掲示板型 SNS (Reddit) のコメント文を収集	複数人合意に基づく感情ラベル	BERT [†] 等
MERR ⁽²²⁾	28k (+4.5k)	感情認識コンテスト用の映画やドラマの動画画像	OpenFace, MiniGPT 等の自動評価	LLaMA [†] 等
DeepMood ⁽²⁸⁾	2,382 人	行動ログ（起床時刻や外出有無等）をスマホ申告	気分（良好、憂鬱等）の自己評価	LSTM
StudentLife ⁽²⁹⁾	48 人	行動ログ（GPS, 加速度, 睡眠データ等）を自動収集	感情の自己評価や学業成績等	線形回帰等
WESAD ⁽²⁷⁾	15 人	生体信号や加速度等を手首と胸のデバイスで取得	中立／ストレス／喜びの自己評価	Random Forest

† BERT と LLaMA は、それぞれ Google と Meta が開発した Transformer ベースのモデルで、BERT は主に自然言語理解、LLaMA は文生成に特化した大規模言語モデルである。

グごとに内面状態を回答させる。

・他者評価 当事者以外の第三者の人間がラベルを付与する方法である。Affectnet⁽²⁶⁾やGoEmotions⁽²⁴⁾などの表情画像やSNS投稿文に用いられ、一貫性を保つために複数人で意見を照合することが多い。ここで言う第三者とは、Parkinsons⁽³⁰⁾のように医師などの専門家である場合もあれば、クラウドソーシングにより募った不特定多数のワーカー⁽³¹⁾である場合もある。

・自動評価 各分野の知見（例えば、感情語辞書）や大規模言語モデルなどを用いる方法である。手動での評価が必要な先の二つと比べると大きく労力を減らせるが、知見や技術の性能に大きく左右され、人の判断と整合しないこともある。近年 MERR⁽²²⁾のように、画像処理や言語モデルを組み合わせた成功事例も生まれている。ラベルの品質はモデルの性能に大きく関わり、いずれの方法を用いる場合でも品質管理は重要となる。

2.2 内面状態推定の定式化

上記の手続きにより得られたラベル付きデータ $\mathcal{D}$ を用いて計測値 x と内面状態のラベル y の関係を学習する。これは、条件付き確率分布 $p(y|x;w)$ をパラメータ w を持つモデルで表現し、パラメータを推定する問題として定式化される。ラベル y が連続値の場合には正規分布 $\mathcal{N}$ 、ラベル y が離散値（0が良好、1が高ストレス等）の場合にはシグモイド関数 S を用いた以下の形式のモデルを利用することが多い。

$$\text{回帰 } (y \in \mathbb{R}) : p(y|x;w) = \mathcal{N}(y; h_w(x), \sigma^2), \quad (1)$$

$$\text{識別 } (y \in \{0, 1\}) : p(y=1|x;w) = S(h_w(x)). \quad (2)$$

パラメータに依存する項 $h_w(x)$ には、古典的な線形モデル $h_w^{\text{LM}}(x) = w^T x$ やConvolutional Neural Network (CNN), Recurrent Neural Network (RNN), Long Short Term Memory (LSTM), Transformerなどをベースとした深層学習モデルが用いられる^{(19), (28), (32)}。モデルのパラメータ w は、データ $\mathcal{D}$ を用いて定義される対数ゆう度関数 $\mathcal{L}$ の最大化により得ることができる。

$$\hat{w} = \arg \max_w \mathcal{L}(w), \quad \mathcal{L}(w) = \sum_{i=1}^N \log p(y_i | x_i; w). \quad (3)$$

対数ゆう度関数(式(3))の最大化は、式(1)の場合は二乗誤差の最小化、式(2)の場合はクロスエントロピー損失の最小化と等価である。最適化手法には、最小二乗法や確率的勾配降下法など、利用するデータの量やモデルの種類に適した手法が採用される。

2.3 内面状態推定の難しさ

ここまで説明した流れで、内面状態推定のためのモデルを構築できる。しかし、本質的に対象が人間であることに起因した下記の課題などから、十分な性能が得られなかったり、実用の現場に適用するのが困難になる。

課題1 正確なアノテーションが困難：内面状態は主観的であり、自己評価による方法は時間や文脈に依存して一貫性に欠けた回答となる可能性がある。複数人の他者評価であったとしても判断が割れることもあり、ラベルを付与する評価者のバイア

スを考慮して取り扱う必要がある。

課題2 計測情報の収集が困難：Web や SNS 上のものが利用できる一部の場合を除き、計測情報の取得にはプライバシーや倫理的な配慮の上、計測対象者の協力を仰ぐ必要があり、大規模・長期間のデータ計測は難しい。一部の行動ログや生体信号は、スマートフォンや腕時計型などのウェアラブルデバイスの普及により、比較的少ないユーザ負荷で常時計測可能となったが、心電位や脳波のように専用機器を使って計測される情報と比べると、取得できる情報の粒度や品質に課題がある⁽³³⁾。

課題3 個人差の存在：人の内面状態は極めて個人差が大きく、例えばある人にとってはストレスレベルの高い行動ログ上のパターンが、ほかの人には当てはまらないことがある。また、複数人に全く同じ映像を提示しデータを収集する場合にも、誘発される感情が人ごとに異なり得ることから、ある感情状態のラベルが一部の人からは集められないこともある。このような計測情報と内面状態の関係性や、内面状態のラベル分布に存在する個人差がモデルの性能を高める上での障壁となる。

課題1 に対しては、研究領域ごとに関連する様々な試みがある。例えば、HCI やヒューマンコンピューターションの分野では、経験サンプリング法における自己評価の時間変化の調査⁽³⁴⁾や、複数人のクラウドワーカーによるアノテーションを評価・管理するための研究が進められている⁽³¹⁾。ほかにも感情計算論の文脈で、回答バイアスを除去する手法^{(35), (36)}やランキング回答（一対比較）を用いたアプローチの検討が進められている⁽³⁷⁾。

課題2 と課題3 に対して有望なアプローチが、次章で述べる拡散モデルを用いる方法である。拡散モデルが持つ高いデータ生成能力を活用し、スマートウォッチで計測可能なデータを専用デバイスが必要なデータに変換するモダリティ変換や、量が不足する特定の内面状態ラベルに関するデータを生成し、量を増やすデータ拡張により課題に対処できる可能性がある。

3 内面状態推定における拡散モデル

この章からは内面状態推定における拡散モデルの応用について紹介する。まず、拡散モデルの深層生成モデルにおける位置付けと、データ拡張・モダリティ変換への応用事例について紹介した後、代表的な拡散モデルである DDPM⁽³⁾ と注目のフローモデルである RF⁽⁶⁾ について説明する。最後に脈波から

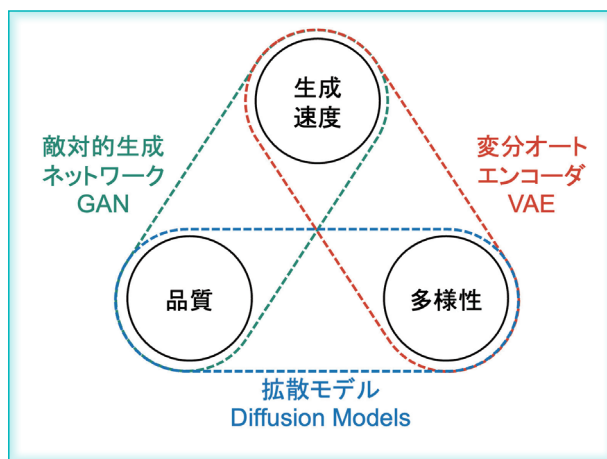


図1 代表的な深層生成モデルの比較図^{(41), (42)}

心電位を生成するモダリティ変換への適用例^{(16), (17)}について示す。

拡散モデルの位置付け：拡散モデルは、Variational Autoencoders (VAE)⁽³⁸⁾ と Generative Adversarial Nets (GAN)⁽³⁹⁾ に並ぶ深層生成モデルの一つとして位置付けられる。VAE と GAN の解説はほかの記事（例えば、文献⁽⁴²⁾, ⁽⁴⁰⁾）に譲るが、各手法の特徴・利点は図1^{(41), (42)}のとおりである。VAE は確率的な潜在変数を利用し多様性に富んだ生成が可能である一方、画像のシャープさに欠けるなど生成されるデータの品質が高くない傾向にある。GAN は生成器と識別器が対戦することで高品質な画像を生成可能だが、学習が安定せず「モード崩壊」と呼ばれる、一部の典型的なデータしか生成できない問題が知られている。これに対し、拡散モデルは生成速度に難点はある（後述する）が、高品質かつ多様性に富んだ生成が可能であり、1. で述べた画像生成などに広く利用されている。なお、図1に示した各手法の欠点については、これを改善する様々な研究が行われていることを明記しておく。例えば、拡散モデルと GAN を組み合わせて欠点を補う手法が提案されている⁽⁴¹⁾。

拡散モデルを用いたデータ拡張：量の不足する学習データを生成することで、内面状態推定モデルの性能を向上させるアプローチである。感情に関するデータを収集する場合は、減多に怒ることのない穏和な人から怒りを感じた際の情報を計測することが難しいなど、計測が困難なラベルが存在する。そこで、拡散モデルを用いて計測が困難なラベルに関する多様なデータを生成し、データセットの質と量を向上させることで、分類モデルの汎化性能や雑音（ノイズ）に対する頑健性が向上することが示されている⁽⁴³⁾。同様に、ある疾患に関するデータを集

める場合にも、疾患を持つ患者や症状が現れる期間が希少なことがあり、拡散モデルによるデータ拡張を用いた推定性能の向上が報告されている^{(44), (45)}。

拡散モデルを用いたモダリティ変換：内面状態との関連は弱い計測が容易なモダリティを、計測は難しいが内面状態との関連が強いモダリティへと変換することで、データ計測の容易さと内面状態推定の性能を両立するアプローチである。例えば、専用機器を使って計測する心電位や脳波は、計測コストが高い代わりに内面状態をよく反映しているとされており、脈波やチャネル数の少ない脳波は、容易に計測可能な代わりにノイズの影響を受けやすく、内面状態の推定に寄与しにくいことが知られている。そこで、脈波から心電位を生成する手法^{(16), (17)}や、チャネル数の少ない脳波計で計測した脳波をチャネル数の多い脳波に変換する手法⁽⁴⁶⁾などの研究が進められている。

3.1 DDPM の定義

ここからは上記事例で利用される DDPM⁽³⁾ の基礎的な原理について説明する。DDPM は、データの生成を「ランダムなノイズから元のデータを段階的に復元する逆変換の問題」として捉えることで実現された手法であり、ノイズを付与する「拡散過程」とノイズを除去する「逆拡散過程」と呼ばれる二つのプロセスから成る。以後、これらのプロセスの説明に利用する画像や心電位などに対応する観測変数 x_0 と潜在変数 $x_{1:T} = \{x_1, \dots, x_T\}$ をまとめて $x_{0:T}$ と書く。

拡散過程：拡散過程は、観測変数 x_0 に対して時刻 $t = 1, \dots, T$ にわたり逐次的にノイズを加えることで $x_{0:T}$ を生成するマルコフ過程として定義される。

$$q(x_{0:T}) = q(x_0) \prod_{t=1}^T q(x_t | x_{t-1}), \quad (4)$$

$$q(x_t | x_{t-1}) = \mathcal{N}(x_t; \sqrt{1 - \beta_t} x_{t-1}, \beta_t I).$$

2 行目の式は $x_t = \sqrt{1 - \beta_t} x_{t-1} + \sqrt{\beta_t} \epsilon$ (ϵ は標準正規分布に従うノイズ $\epsilon \sim \mathcal{N}(\epsilon; 0, I)$) と表現できることから、 β_t はノイズの大きさを定める値と解釈できる ($0 < \beta_t < 1$)。式 (4) 及び正規分布の再生性から、潜在変数 x_t の従う確率分布の閉形式表現を以下のように導くことができる。

$$q(x_t | x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t} x_0, (1 - \bar{\alpha}_t) I). \quad (5)$$

ただし、 $\alpha_t = 1 - \beta_t$, $\bar{\alpha}_t = \prod_{s=1}^t \alpha_s$ と定義した。式 (5) は、時刻 T のときに $\bar{\alpha}_T \approx 0$ となるように $\{\beta_t\}_{t=1}^T$ を設定することで $q(x_T | x_0) \approx \mathcal{N}(x_T; 0, I)$ となり、標準正規分布と同一視できる。これにより、潜在変数 x_T

を正規分布から生成されたノイズとみなすことができる。また、この後の導出で潜在変数 x_{t-1} に関する下記の条件付き確率も、正規分布で表現できることを利用する。

$$q(x_{t-1} | x_t, x_0) = \mathcal{N}(x_{t-1}; \tilde{\mu}(x_t, x_0), \sigma_t^2 I). \quad (6)$$

$$\text{ただし、} \tilde{\mu}(x_t, x_0) = \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})x_t + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t)x_0}{1 - \bar{\alpha}_t}, \sigma_t^2 =$$

$(1 - \alpha_t)(1 - \bar{\alpha}_{t-1})/(1 - \bar{\alpha}_t)$ である。

逆拡散過程：次に逆拡散過程では、先ほどとは逆の方向、すなわち標準正規分布からサンプリングされた潜在変数 x_T をスタートとして、ニューラルネットワークを用いたノイズ除去を繰り返すマルコフ過程を考える。

$$p_\theta(x_{0:T}) = p(x_T) \prod_{t=1}^T p_\theta(x_{t-1} | x_t), \quad (7)$$

$$p_\theta(x_{t-1} | x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \sigma_t^2 I)$$

2 行目中の平均 μ_θ は、パラメータ θ を持つニューラルネットワークの出力を用いて計算される項であり、詳細については後述する。分散については、導出の都合上、式 (6) と同じ値が通常用いられる。式 (7) から潜在変数 $x_{1:T}$ を周辺消去することで、逆拡散過程の導く観測データ x_0 の生成確率が得られる。

$$p_\theta(x_0) = \int p_\theta(x_{0:T}) dx_{1:T}. \quad (8)$$

図 2 (a) に DDPM のグラフィカルモデルを示す。式 (8) の積分は解析的に計算できないことから、パラメータ θ の推定には次節の変分推論の枠組みを利用する。

3.2 DDPM の損失関数の導出

DDPM のパラメータ推定には、以下の「変分下限」(ELBO: Evidence Lower Bound) と呼ばれる量を用いる。

$$\text{ELBO}(\theta; x_0) = \mathbb{E}_{q(x_{1:T}|x_0)} \left[\log \frac{p_\theta(x_{0:T})}{q(x_{1:T}|x_0)} \right] \quad (9)$$

$$= \sum_{t=1}^T \mathbb{E}_{q(x_t, x_{t-1}|x_0)} \left[\log p_\theta(x_{t-1} | x_t) \right] + \text{Const.}$$

ただし、 $\mathbb{E}_q[\cdot]$ は確率分布 q による期待値操作を表す。変分下限は、式 (8) の対数 (対数ゆう度) $\log p_\theta(x_0)$ の下限であり (イエンセンの不等式により確かめられる)、これを最大化することでパラメータが推定できる。

変分下限を目的関数として利用するため、より単純かつ計算に適した表現を求める。式 (9) の最終行の $t(>1)$ 番目の項に、パラメータ θ に依存しない項 $\mathbb{E}_{q(x_t, x_{t-1}|x_0)} [-\log q(x_{t-1} | x_t, x_0)]$ を加えた関数を

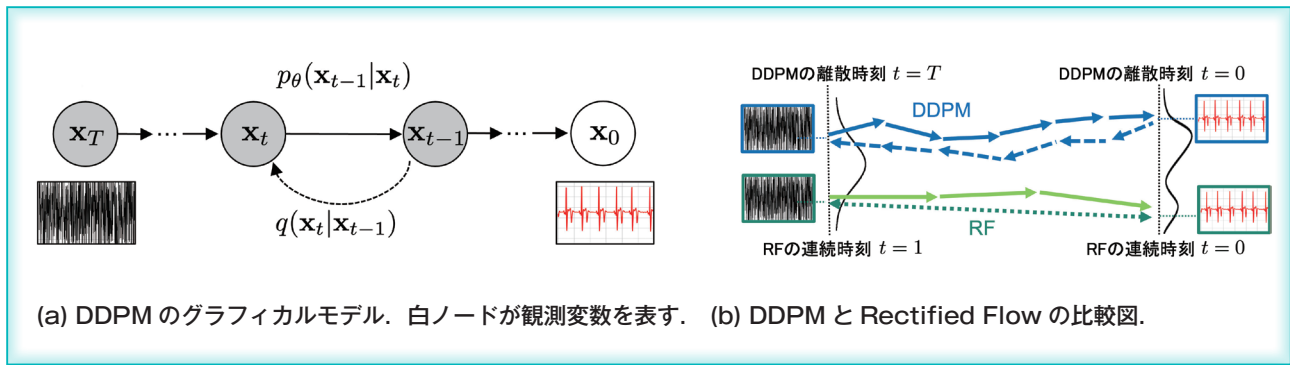


図2 DDPM のグラフィカルモデルと Rectified Flow の比較図

$\mathcal{J}_t(\theta)$ と書く. これは以下のように表現できる.

$$\begin{aligned} \mathcal{J}_t(\theta) &= \mathbb{E}_{q(x_t|x_0)q(x_{t-1}|x_t,x_0)} \left[\log \frac{p_\theta(x_{t-1}|x_t)}{q(x_{t-1}|x_t,x_0)} \right] \\ &= -\mathbb{E}_{q(x_t|x_0)} [\|\mu_\theta(x_t,t) - \tilde{\mu}(x_t,x_0)\|^2] / 2\sigma_t^2 + \text{Const.} \end{aligned} \quad (10)$$

ただし, 最終行の式変形では, 期待値の中の項が分散の等しい正規分布である式 (6) と式 (7) の KL ダイバージェンスとして表せることを利用した. 更に式 (10) の期待値項のノイズ $\epsilon \sim \mathcal{N}(\epsilon; 0, I)$ を用いた表現を導く^{*2}. 式 (5) から, 潜在変数 $x_t \sim q(x_t|x_0)$ は $x_t = \sqrt{\tilde{\alpha}_t}x_0 + \sqrt{1 - \tilde{\alpha}_t}\epsilon \Leftrightarrow x_0 = (x_t - \sqrt{1 - \tilde{\alpha}_t}\epsilon) / \sqrt{\tilde{\alpha}_t}$ と書ける. これを $\tilde{\mu}(x_t,x_0)$ (式 (6)) に代入し整理すると, ノイズ ϵ を用いた表現が得られる.

$$\tilde{\mu}(x_t,x_0) = \frac{1}{\sqrt{\tilde{\alpha}_t}} \left(x_t - \frac{\beta_t}{\sqrt{1 - \tilde{\alpha}_t}} \epsilon \right). \quad (11)$$

この式の形に対応するように $\mu_\theta(x_t,t)$ (式 (7)) を

$$\mu_\theta(x_t,t) = \frac{1}{\sqrt{\tilde{\alpha}_t}} \left(x_t - \frac{\beta_t}{\sqrt{1 - \tilde{\alpha}_t}} \epsilon_\theta(x_t,t) \right) \quad (12)$$

と設計する. ただし, ϵ_θ はニューラルネットワークの出力を表し, U-Net⁽⁴⁷⁾ がベースとされること

*2 再パラメータ化トリック (reparametrization trick) と呼ばれる.

が多い. 式 (11), (12) を用いると, 関数 $\mathcal{J}_t(\theta)$ (式 (10)) の期待値項を以下のように変形できる.

$$\begin{aligned} &\frac{1}{2\sigma_t^2} \mathbb{E}_{q(x_t|x_0)} [\|\mu_\theta(x_t,t) - \tilde{\mu}(x_t,x_0)\|^2] \\ &= w_t \mathbb{E}_{\epsilon \sim \mathcal{N}(\epsilon; 0, I)} [\|\epsilon_\theta(\sqrt{\tilde{\alpha}_t}x_0 + \sqrt{1 - \tilde{\alpha}_t}\epsilon, t) - \epsilon\|^2]. \end{aligned} \quad (13)$$

ただし, $w_t = \beta_t^2 / \{2\sigma_t^2 \alpha_t (1 - \tilde{\alpha}_t)\}$ である. 上記の導出に基づき, DDPM は関数 $\mathcal{L}_{\text{DDPM}}(\theta)$ をパラメータ推定において最小化する損失関数として利用する.

$$\mathcal{L}_{\text{DDPM}}(\theta) = \mathbb{E}_{x_0, \epsilon, t} [\|\epsilon_\theta(\sqrt{\tilde{\alpha}_t}x_0 + \sqrt{1 - \tilde{\alpha}_t}\epsilon, t) - \epsilon\|^2]. \quad (14)$$

ただし, $\mathbb{E}_{\epsilon, t, x_0}[\cdot]$ はノイズ $\epsilon \sim \mathcal{N}(\epsilon; 0, I)$, 離散の一樣分布に従う時刻 $t \sim \text{Uniform}(\{1, \dots, T\})$, データからランダム選択する x_0 の出方に関する期待値を表す. また, 式 (13) における係数 w_t は無視しても同等の結果が得られたとの報告⁽³⁾ に基づき $w_t = 1$ と設定される.

3.3 DDPM の学習とデータ生成

学習: 変分下限から導いた上記の損失関数 (式 (14)) は, x_0 から x_t を作る時に使うノイズ ϵ の値を予測する, ノイズ予測問題の損失関数として解釈できるものである. したがって, Algorithm 1 (図 3) に示す擬似コードのように, 勾配法ベースの最適化手法を用いて DDPM の学習 (パラメータ

Algorithm 1 DDPM と RF の学習アルゴリズム

Input: 学習データ $\mathcal{D}_U$, モデル ϵ_θ (or v_θ) **Output:** パラメータ θ

- 1: **while** 終了条件を満たさない **do**
- 2: /* データとノイズ, 時刻 t のサンプリング */
- 3: $(x_0, c) \sim \mathcal{D}_U$, $\epsilon \sim \mathcal{N}(\epsilon; 0, I)$, $t \sim \text{Uniform}([0, 1])$
- 4: $t' = \lfloor t(T-1) \rfloor + 1$ /* 時刻を離散化 */
- 5: /* データ x_0 にノイズを付与し $x_{t'}$ を作成 */
- 6: (DDPM) $x_{t'} \leftarrow \sqrt{\tilde{\alpha}_{t'}}x_0 + \sqrt{1 - \tilde{\alpha}_{t'}}\epsilon$
- 7: (RF) $x_t \leftarrow (1-t)x_0 + t\epsilon$
- 8: /* ノイズ予測/ベクトル場帰帰の誤差を最小化 */
- 9: (DDPM) 勾配 $\frac{\partial}{\partial \theta} \|\epsilon_\theta(x_{t'}, t', c) - \epsilon\|^2$ を使い θ を更新
- 10: (RF) 勾配 $\frac{\partial}{\partial \theta} \|v_\theta(x_t, t, c) - (\epsilon - x_0)\|^2$ を使い θ を更新
- 11: **end while**

図3 DDPM と RF の学習アルゴリズム

Algorithm 2 生成アルゴリズム

Input: 条件 c , 生成ステップ数 T **Output:** 生成した心電位 x_0

```

1:  $x_T \leftarrow \epsilon \sim N(\epsilon; 0, I)$  /* 初期値をサンプリング */
2: for  $t = T, \dots, 1$  do
3:   /*  $z$  でランダム性を持たせながらノイズ除去 */
4:    $z \leftarrow N(z; 0, I)$  if  $t \neq 1$ , else  $z = 0$ 
5:   (DDPM):  $x_{t-1} \leftarrow \frac{1}{\sqrt{\alpha_t}} (x_t - \frac{\beta_t}{\sqrt{1-\alpha_t}} \epsilon_\theta(x_t, t, c)) + \sigma_t z$ 
6:   /* 所望のデータに至る勾配を推定して遷移 */
7:   (RF):  $x_{t-1} \leftarrow x_t - \frac{1}{T} v_\theta(x_t, \frac{t}{T}, c)$ 
8: end for
```

図4 生成アルゴリズム

θ の推定)が可能となる。

データ生成: DDPMによるデータ生成には逆拡散過程(式(7))を利用する。Algorithm 2(図4)に示すとおり、正規分布からサンプリングした初期値のノイズ ϵ に対して、各ステップ t ごとに $p_\theta(x_{t-1}|x_t)$ に従うノイズ除去、すなわち平均が μ_θ (式(12))の正規分布からのサンプリングを繰り返すことでデータを生成する。各ステップでサンプリング操作を行うことから、同じ初期値に対しても出力が異なる多様なデータが生成できる。

DDPMによるデータ生成は前述のとおり画像生成で大きな成功をもたらしたものの、生成時にノイズ除去の操作を複数回(通常 $T=200 \sim 1000$ 回程度)行う必要がある。そのため、データの生成には通常の深層学習モデルの推論時間の数百倍の時間が掛かるという課題が広く認識され、研究が進められている^{(6), (8), (9)}。

3.4 Rectified Flow

DDPMのデータ生成に関する課題に対応する手法として注目されているのが、フローモデルのRF⁽⁶⁾である。RFは、ノイズを初期値としてデータを生成する点はDDPMと共通である。しかし、離散時間上の確率的な拡散過程を利用するDDPMとは異なり、RFは連続時間上の決定的な常微分方程式(ODE)を利用しノイズからデータへ変換するベクトル場を学習する手法である。以下で具体的に説明する。

時刻 $t=0$ のデータ x_0 とそれに対応する時刻 $t=1$ のノイズ x_1 の2点を結ぶ直線 $x_t=(1-t)x_0+tx_1$ ($t \in [0, 1]$)を考える。RFは、これがノイズからデータ(及びデータからノイズ)への変換の軌道を表すものであると仮定する(図2(b)の緑点線)。この直線の傾き v は時刻 t によらず一定値 $v=x_1-x_0$ となることを利用し、RFでは、直線上の任意の点 x_t から、この傾き v を予測するベクトル場モデル $v_\theta(x_t, t)$ を

学習する。損失関数には以下の関数 $\mathcal{L}_{\text{RF}}$ が利用される。

$$\mathcal{L}_{\text{RF}}(\theta) = \mathbb{E}_{x_0, \epsilon, t} [\|v_\theta((1-t)x_0 + t\epsilon, t) - v\|^2], \quad (15)$$

この損失関数(式(15))は、変分下限から導かれたDDPMのノイズ予測損失関数(式(14))とは異なり、ベクトル場(フロー)を推定するための損失関数として直接導入されている。DDPMと同様に(連続値の)時刻 t とノイズ ϵ のサンプリング及び勾配ペースの最適化を行うアルゴリズム1でパラメータを推定することができる。

RFにおけるデータ生成は、ノイズからデータへの変換が直線で表現できるとした仮定に基づけば、理想的には1ステップで実行できるが、学習されたモデルの出力 $v_\theta(x_t, t)$ が任意の時刻 t で v と等しくなる(変換が直線となる)ことは保証されない。そこでデータ生成の際には、常微分方程式(ODE)の初期値問題

$$\dot{x}_0 = x_1 - \int_0^1 v_\theta(x_1, t) dt, \quad (16)$$

をオイラー法などのODEソルバにより解くことでデータ $\hat{x}_0$ を生成する(図2(b)の黄緑実線)。訓練時に設定したステップ数と推論時のステップ数が同一となる必要があるDDPMと異なり、RFではデータ生成時に離散化するステップ数 T を適応的に設定することが可能である(Algorithm 2, 図2)。これによりDDPMの課題であったステップ数の削減を可能としている。実用上は $T=5 \sim 50$ 設定されることが多い^{(14), (15)}。これまでの説明を踏まえ、DDPMとRFの典型的な設定の違いは表3にまとめた。

3.5 脈波から心電位へのモダリティ変換

条件付き拡散モデル: DDPMやRFなど多くの生成モデルでは、モデルの入力に条件付け(クラスラベルやテキストなど)を与えることで、生成するデータを指定することができる^{(13), (14)}。例えば、条件

表3 DDPM と RF の比較表

比較項目	DDPM	RF
目的関数	$\ \epsilon_{\theta}(x_t, t, c) - \epsilon\ ^2$	$\ v_{\theta}(x_t, t, c) - (\epsilon - x_0)\ ^2$
生成の確率性	確率的	決定的
ステップ数	200 ~ 1000 程度	5 ~ 50 程度
典型事例	初期検討として適用	既存拡散モデルの高速化

付けとしてテキストを与えることで、テキストに合致した画像の生成が可能でありサービスの普及が進んでいる。この条件付けとして、心電位と同時刻に計測された脈波を与えることで、脈波から心電位を生成するモデル変換が実現できる。具体的には、時刻同期された脈波と心電位のペアのデータセット $\mathcal{D}_U = \{x_{0,i}, c_i\}_{i=1}^n$ が与えられたとき、Algorithm 1, 2 に示したようにモデル $\epsilon_{\theta}, v_{\theta}$ の入力に脈波 c_i を追加することで、脈波で条件付けられた心電位 $\hat{x}_{0,i}$ の生成が可能となる。

心電位生成への特化：DDPM と RF は汎用的な生成モデルであり、脈波から心電位を生成することに特化したモデルではない。そこで、脈波から心電位を生成することにそれぞれを特化させた手法として、RDDM⁽¹⁶⁾ と我々の取り組みである CardioFlow⁽¹⁷⁾ がある。いずれの手法も、心電波形の重要な特徴であるピーク（R 波）領域を詳細に再現することに焦点を当てている。

DDPM の拡張手法である RDDM⁽¹⁶⁾ では、心電位全体にノイズを加える拡散過程と、R 波の近傍領域にのみノイズを加える拡散過程の二つのプロセスを考える。そして、それぞれの拡散過程に対する逆拡散過程を実現する異なる二つのニューラルネットを学習する。

RF を拡張した手法である CardioFlow⁽¹⁷⁾ では、R 波領域の生成を促すために毎エポックごとに一定の確率で、心電位の R 波の近傍領域以外と脈波のピークの近傍領域以外をマスクする処理を追加する。

$$x_{0,i} \leftarrow x_{0,i} \cdot m_i^{\text{ECG}}, \quad c_i \leftarrow c_i \cdot m_i^{\text{PPG}}.$$

ここで、 $m_i^{\text{ECG}}, m_i^{\text{PPG}}$ はそれぞれ i 番目の心電位と脈

波のピーク領域を 1、それ以外を 0 としたマスクデータである。これは、脈波と心電位に現れるピークには相関があり、心電位のピークから遅れて脈波にピークが現れるという医学的知識を活用して、ピーク領域のみを持つ脈波から、R 波領域のみを持つ心電位の生成を期待するものである。いずれの手法も、モデルを訓練する前に既知の脈波・心電位のピーク検出アルゴリズムを適用することでピーク位置を算出して、ノイズ付加やマスクを施している。

心電位の生成例：図 5 に DDPM, RDDM, RF, CardioFlow により生成した心電位を示す。これらの心電位は、WESAD データセット⁽²⁷⁾ に含まれる 15 名の脈波・心電位データのうち、9 名のデータで訓練、3 名のデータで検証したモデルを、残りの 3 名に適用した結果の一部である。心電位生成品質の評価指標には、平均二乗誤差や Fréchet distance, 1 分当りの心拍数の絶対誤差が用いられることが多い。図 5 から、心電位生成に特化した RDDM と CardioFlow の方がピークの数が正確であり、高い生成品質を持つことが分かる。更に、RDDM や CardioFlow により生成した心電位を訓練データとして感情認識モデルを学習することで、変換前の脈波を用いたモデルよりも高い性能となることが確認されており、定性面・定量面の両方でモデル変換手法としての有用性が示されている^{(16), (17)}。

4 まとめ

本稿では、主に機械学習の観点から内面状態推定の基礎、及び DDPM・RF によるデータ生成の原理



図5 脈波から生成した心電位の例 各行は異なる三つの生成例を示す。入力データには、128 Hz にリサンプリングした脈波と心電位を 4 秒単位で分割したのち、min-max 正規化を施したものをを用いた。

と応用事例について解説した。今後、実環境への適用や高精度なパーソナライズ化を進める上では、拡散モデルを含む生成的アプローチを活用することが、課題を乗り越える鍵となることが予想される。適用先のデータやモダリティによっては、既存の手法をそのまま適用するだけでは十分な品質のデータ生成が困難な場合もあると考えられるが、心電位へのモダリティ変換の例で示したように、適用先のドメイン知識やデータ固有の特徴を活用することで対応可能である、と筆者らは考える。本稿が内面状態推定に関心を持つ読者の研究や応用の一助となれば幸いである。拡散モデルを含む深層生成モデルや機械学習とは異なる観点等に興味がある読者は、本文中で引用したサーベイ論文・書籍（日本語の文献(4)、(31)、(36)、(40)、(42)）についても参照されたい。

■文献

- (1) 三菱総合研究所, “体の情報から感情を推定するデジタル技術が進展,” Jan. 2024. <https://www.mri.co.jp/knowledge/column/20240124.html>
- (2) NTT 人間情報研究所共生知能研究プロジェクト, “NTT SYMBIOTICINTELLIGENCE LABORATORY White Paper,” 2025. <https://www.rd.ntt/hil/category/emotion/>
- (3) J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” *Advances in neural inf. Process. Syst.*, vol. 33, pp. 6840-6851, 2020.
- (4) 岡野原大輔, 拡散モデル: データ生成技術の数理, 岩波書店, 東京, 2023.
- (5) I. Kobyzev, S. J. Prince, and M. A. Brubaker, “Normalizing flows: An introduction and review of current methods,” *IEEE trans. pattern analysis and mach. Intel.*, vol. 43, no. 11, pp. 3964-3979, May 2020.
- (6) X. Liu, C. Gong, and Q. Liu, “Flow straight and fast: Learning to generate and transfer data with rectified flow,” *International Conf. Learning Representations*, 2023.
- (7) Y. Lipman, R. T. Chen, H. Ben-Hamu, M. Nickel, and M. Le, “Flow matching for generative modeling,” *International Conf. Learning Representations*, 2023.
- (8) J. Song, C. Meng, and S. Ermon, “Denoising diffusion implicit models,” *International Conf. Learning Representations*, 2021.
- (9) C. Lu, Y. Zhou, F. Bao, J. Chen, C. Li, and J. Zhu, “Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps,” *Advances in neural inf. Process. syst.*, vol. 35, pp. 5775-5787, Nov. 2022.
- (10) C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans, J. Ho, D. J. Fleet, and M. Norouzi, “Photorealistic text-to-image diffusion models with deep language understanding,” *Advances in neural inf. processing syst.*, vol. 35, pp. 36479-36494, Nov. 2022.
- (11) A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, “Hierarchical text-conditional image generation with clip latents,” *arXiv preprint arXiv:2204.06125*, vol. 1, no. 2, p. 3, April 2022.
- (12) O. Avrahami, D. Lischinski, and O. Fried, “Blended diffusion for text-driven editing of natural images,” *IEEE/CVF conf. comput. vision and pattern recognition*, pp. 18208-18218, June 2022.
- (13) R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-resolution image synthesis with latent diffusion models,” *IEEE/CVF Conf. Comput. Vision and Pattern Recognition*, pp. 10684-10695, June 2022.
- (14) P. Esser, S. Kulal, A. Blattmann, R. Entezari, J. Müller, H. Saini, Y. Levi, D. Lorenz, A. Sauer, F. Boesel, D. Podell, T. Dockhorn, Z. English, and R. Rombach, “Scaling rectified flow transformers for high-resolution image synthesis,” *International conf. machine learning*, July 2024.
- (15) B. Kim, Y.-G. Hsieh, M. Klein, M. Cuturi, J. C. YE, B. Kavar, and J. Thornton, “Simple reflow: Improved techniques for fast flow models,” *International Conf. Representation Learning*, pp. 67740-67766, 2025.
- (16) D. Shome, P. Sarkar, and A. Etemad, “Region-disentangled diffusion model for high-fidelity ppg-to-ecg translation,” *AAAI conf. artificial intel.*, vol. 38, pp. 15009-15019, 2024.
- (17) Y. Nambu, M. Kohjima, and R. Yamamoto, “Cardioflow: Learning to generate ecg from ppg with rectified flow,” *IEEE International Conf. Acoustics, Speech and Signal Process.*, pp. 1-5, 2025.
- (18) Y.-L. Tian, T. Kanade, and J. F. Cohn, “Recognizing action units for facial expression analysis,” *IEEE trans. pattern analysis and machine intel.*, vol. 23, no. 2, pp. 97-115, 2001.
- (19) S. Li and W. Deng, “Deep facial expression recognition: A survey,” *IEEE trans. affective comput.*, vol. 13, no. 3, pp. 1195-1215, 2020.
- (20) D. Avola, L. Cinque, A. Fagioli, G. L. Foresti, and C. Massaroni, “Deep temporal analysis for non-acted body affect recognition,” *IEEE Trans. Affective Comput.*, vol. 13, no. 3, pp. 1366-1377, 2020.
- (21) Y. Fukazawa, T. Ito, T. Okimura, Y. Yamashita, T. Maeda, and J. Ota, “Predicting anxiety state using smartphone-based passive sensing,” *J. biomedical informatics*, vol. 93, p. 103151, 2019.
- (22) Z. Cheng, Z.-Q. Cheng, J.-Y. He, K. Wang, Y. Lin, Z. Lian, X. Peng, and A. Hauptmann, “Emotion-llama: Multimodal emotion recognition and reasoning with instruction tuning,” *Advances in Neural Inf. Process. Syst.*, vol. 37, pp. 110805-110853, 2024.
- (23) P. Ekman, “An argument for basic emotions,” *Cognition & emotion*, vol. 6, no. 3-4, pp. 169-200, 1992.
- (24) D. Demszky, D. Movshovitz-Attias, J. Ko, A. Cowen, G. Nemade, and S. Ravi, “Goemotions: A dataset of fine-grained emotions,” *Annual Meeting of the Association for Computational*

- Linguistics, pp. 4040-4054, 2020.
- (25) H. Schlosberg, "Three dimensions of emotion," Psychological review, vol. 61, no. 2, p. 81, 1954.
- (26) A. Mollahosseini, B. Hasani, and M. H. Mahoor, "Affectnet: A database for facial expression, valence, and arousal computing in the wild," IEEE Trans. Affective Computing, vol. 10, no. 1, pp. 18-31, 2017.
- (27) P. Schmidt, A. Reiss, R. Duerichen, C. Marberger, and K. Van Laerhoven, "Introducing wesad, a multimodal dataset for wearable stress and affect detection," ACM international conf. multimodal interaction, pp. 400-408, 2018.
- (28) Y. Suhara, Y. Xu, and A. Pentland, "Deepmood: Forecasting depressed mood based on self-reported histories via recurrent neural networks," International Conf. World Wide Web, pp. 715-724, 2017.
- (29) R. Wang, F. Chen, Z. Chen, T. Li, G. Harari, S. Tignor, X. Zhou, D. Ben-Zeev, and A.T. Campbell, "Studentlife: assessing mental health, academic performance and behavioral trends of college students using smartphones," ACM international joint conf. pervasive and ubiquitous computing, pp. 3-14, 2014.
- (30) A. Tsanas, M. Little, P. McSharry, and L. Ramig, "Accurate telemonitoring of parkinson's disease progression by non-invasive speech tests," Nature Precedings, pp. 1-1, 2009.
- (31) R.M. Monarch, 上田隼也, 角野為耶, 伊藤寛祥, Human-in-the-Loop 機械学習: 人間参加型 AI のための能動学習とアノテーション, 共立出版, 2023.
- (32) J. Wagner, A. Triantafyllopoulos, H. Wierstorf, M. Schmitt, F. Burkhardt, F. Eyben, and B. W. Schuller, "Dawn of the transformer era in speech emotion recognition: closing the valence gap," IEEE Trans. Pattern Analysis and Machine Intell., vol. 45, no. 9, pp. 10745-10759, Sept. 2023.
- (33) S. Huhn, M. Axt, H.-C. Gunga, M.A. Maggioni, S. Munga, D. Obor, A. Si'e, V. Boudo, A. Bunker, R. Sauerborn, et al., "The impact of wearable technologies in health research: scoping review," JMIR mHealth and uHealth, vol.10, no.1, p. e34384, 2022.
- (34) T. Tominaga, S. Yamamoto, T. Kurashima, and H. Toda, "Effects of personal characteristics on temporal response patterns in ecological momentary assessments," IFIP Conf. Human-Computer Interaction, pp. 3-22, 2021.
- (35) S. Kumano and K. Nomura, "Multitask item response models for response bias removal from affective ratings," IEEE Conf. Affective Computing and Intell. Interaction, pp. 1-7, 2019.
- (36) 熊野史朗, "主観感情推定の研究動向," 人工知能, vol. 36, no. 1, pp. 13-20, 2021.
- (37) G. N. Yannakakis and H. P. Martínez, "Ratings are overrated!," Frontiers in ICT, vol. 2, p. 13, 2015.
- (38) D. P. Kingma and M. Welling, "Auto-encoding variational bayes," International Conf. Learning Representations, 2014.
- (39) I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," Advances in neural inf. processing syst., vol. 27, 2014.
- (40) 岩田具治, "確率的潜在変数モデルに基づくデータマインニング," 日本オペレーションズ・リサーチ学会機関誌, vol. 64, no. 5, pp. 272-277, 2019.
- (41) Z. Xiao, K. Kreis, and A. Vahdat, "Tackling the generative learning trilemma with denoising diffusion gans," International Conf. Representations, 2022.
- (42) 金子卓弘, "基礎から一歩進んだ深層学習トピックス—ii—gan の基礎・発展・応用," システム／制御／情報, vol. 68, no. 4, pp. 154-159, 2024.
- (43) M. I. Malik, S. Latif, R. Jurdak, and B. W. Schuller, "A preliminary study on augmenting speech emotion recognition using a diffusion model," Interspeech, pp. 646-650, 2023.
- (44) Y. Wang, S. Zhao, H. Jiang, S. Li, B. Luo, T. Li, and G. Pan, "Diffmdd: A diffusion-based deep learning framework for mdd diagnosis using eeg," IEEE Trans. Neural Syst. and Rehabilitation Eng., vol. 32, pp. 728-738, 2024.
- (45) N. Neifar, A. Ben-Hamadou, A. Mdhaffar, and M. Jmaiel, "Diffecg: A versatile probabilistic diffusion model for ecg signals synthesis," IEEE/ACIS Int. Conf. Software Eng. Res., Management and Appl., pp. 182-188, 2024.
- (46) S. Wang, T. Zhou, Y. Shen, Y. Li, G. Huang, and Y. Hu, "Generative ai enables eeg super-resolution via spatio-temporal adaptive diffusion learning," IEEE Trans. Consumer Electron., vol. 71, no. 1, pp. 1034-1045, 2025.
- (47) O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," Medical Image Comput. and Computer-Assisted InterventionSpringer, pp. 234-241, 2015.

(2025 年 8 月 1 日受付, 10 月 1 日再受付)

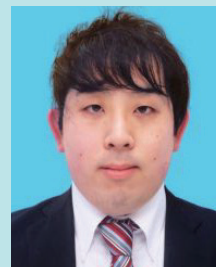
幸島匡宏 (正員)

2009 東工大・工・情報卒。2012 同大学院総合理工学研究科知能システム科学専攻修士課程了。同年 NTT 入社。2019 東工大情報理工学院数理・計算科学系博士課程了。博士(理学)。現在, NTT 人間情報研究所主任研究員。機械学習, 特に確率モデル・不完全データ分析の研究に従事。



南部優太

2019 はこだて未来大・システム情報科学卒。2021 同大学院システム情報科学研究科システム情報科学専攻修士課程了。同年 NTT 入社。2025 はこだて未来大システム情報科学研究科博士課程了。博士(情報科学)。現在, NTT 人間情報研究所研究員。機械学習, 特に生成モデル・不完全データ分析の研究に従事。



5GC環境における、ネットワークデジタルツインを用いたAIモデル構築手法の検討

Study of AI Model Training Method Using Network Digital Twin in 5GC Environment

小山大輝 Daiki Koyama[†] 桜庭皆人 Minato Sakuraba[†]
河崎純一 Junichi Kawasaki[†] 宮坂拓也 Takuya Miyasaka[†]

Summary

本稿では、5GC 環境におけるネットワークデジタルツインを活用した、ネットワーク運用のための AI モデル構築手法に関して報告する。ネットワークデジタルツイン環境において、AI モデル構築のために最低限のスケールアウトされた 5GC NF を構築することで、ネットワークデジタルツインに必要な設備リソース量を抑えつつ、商用ネットワーク上では実施が困難な試験を実施し、試験を通じて得られるデータセットを基に AI モデルを構築する。free5GC と Kubernetes を用いたスケールアウト 5GC 環境において、異なるネットワーク規模 (pod 数) 間の AI モデル適用に関する評価を実施し、その結果について報告する。

Key Words

AI Ops, Autoencoder, 障害検知, 5GC, ネットワークデジタルツイン

1 はじめに

ネットワーク障害への早急な対処は、高品質な通信サービスを継続的にユーザへ提供するために、ネットワーク運用において欠かせない要素である。そのようなネットワーク運用を実現するためには、人間の運用者を介さずに、ネットワーク障害へ対処する自律的なネットワーク運用が求められ、近年技術革新の目覚ましい AI/ML 技術を活用したネットワーク運用の研究開発が進められている⁽¹⁾。そのような中で、我々はモバイルネットワーク (例: 5G, 6G) における AI/ML 技術を用いた自律的なネットワーク運用を目指している。

その上で、人間の運用者と同等以上のネットワーク運用を実現するためには、高精度な AI モデル構築が必要不可欠となり、実際に商用ネットワーク上で構築されている各ネットワーク機器から収集される運用データを用いて、AI モデルが訓練されている。また、そのような運用データには様々な種類があり、時系列数値データ (例: CPU 利用率) や、イベントデータ (例: Syslog) などがある。

しかしながら、AI モデルの利用用途 (例: 障害検知, 原因分析, 将来予測) によっては、商用ネットワーク上で収集が困難なデータが求められる場合

がある。例えば、発生したネットワーク障害種別 (例: CPU 障害, メモリ障害, ネットワーク障害) の特定を行う AI モデルを訓練するためには、各ネットワーク障害時における運用データが、ある程度の数必要となるが、商用ネットワーク上でネットワーク障害は頻繁に発生しないため、各ネットワーク障害時における運用データを十分に収集できないことが考えられ、その結果 AI モデルの精度が劣化してしまう恐れがある。

このような問題を解決するために、商用ネットワークを仮想環境上で模擬したネットワークデジタルツイン (NDT: Network Digital Twin) の適用が検討されている^{(2), (3)}。上記の例においては、商用ネットワークを模擬した NDT 上で、各種ネットワーク障害を意図的に発生させ、商用ネットワークでは収集が困難な運用データを多量かつ迅速に収集し、高精度な AI モデルの訓練が可能となる。しかしながら、商用と完全に同規模の NDT を構築するためには、商用ネットワークと同規模の設備 (例: サーバ, ルータ) を準備する必要があるが、これは単純に通信事業者の設備コストが倍になるという課題がある。

その問題を解決するために、我々はモバイルネットワーク環境 (5GC: 5G Core Network) において、図 1 で示されるような、商用環境上で利用されるスケールアウトされた 5GC Network Function (NF) において、単体の NF 当りのトラヒック負荷 (例: 単位時間当りのリクエスト量) を

[†] (株) KDDI 総合研究所, ふじみ野市
KDDI Research, Inc., Fujimino-shi, 356-8502
Japan

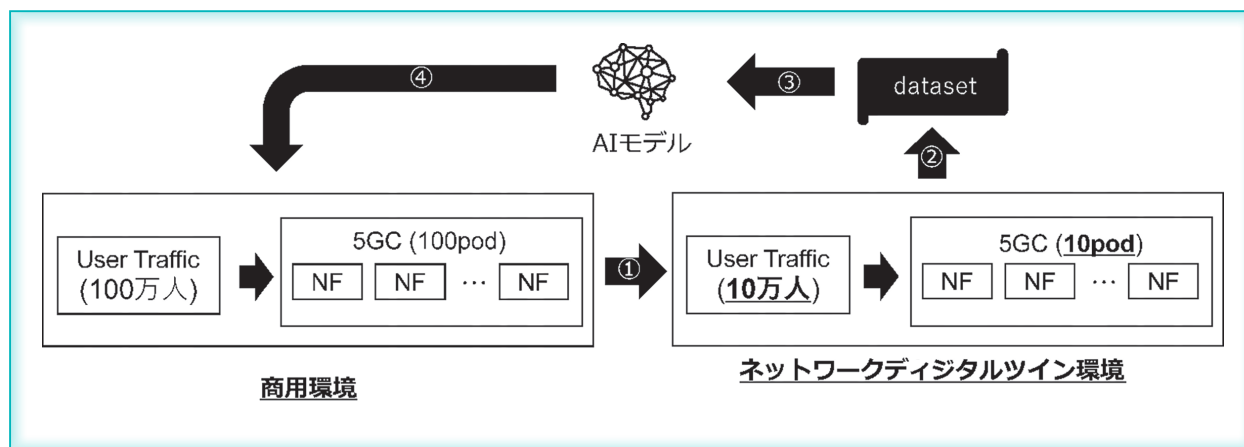


図1 ネットワークデジタルツインを用いた AI モデル構築

同一とする、最低限の数だけNFをスケールアウトした環境をNDTとして利用することで、NDTに要する設備コストを最低限にしながら、高精度なAIモデルの構築が可能となるAIモデル構築手法を提案している。

本手法は、5Gから導入されたService-Based Architecture (SBA) というクラウドネイティブなアーキテクチャに依存している。SBAにおいては、多数のユーザリクエストを処理するために、均一なネットワークインスタンスを並列に多数並べ、ユーザリクエストをそれらインスタンスへ均一に負荷分散させるという、スケールアウト構成が検討されている^{(4)~(7)}。スケールアウト構成が5GCに導入されることにより、図1のようにNFを横並びとする構成が可能となる。NDTにおいて商用ネットワークと完全に同じソフトウェアイメージを用いながら、スケールアウトする要素数を削減し（例：1/10のコンテナ数で構築）、その環境に流れるユーザトラフィックを同様のスケールで低下させることで（例：トラフィック量を全体的に1/10とする。）、単一のノード単位レベルでは商用ネットワークと同様な環境が再現できる。

提案手法の有効性の評価のために、評価用のスケールアウト5GC環境の構築を進めている。本評価環境においては、5GCとしてfree5GC⁽⁸⁾を用い、スケールアウト環境構築のためにKubernetesを用いている。本稿では、初期評価として、本評価環境においてNFのPod数を1~9個まで変更した5GC環境を構築し、それぞれの環境で構築したAIモデルを、各環境に適用した場合のネットワーク障害の検知に関する精度を評価し、その結果を報告する。

2 関連研究：AIによるネットワーク運用

2.1 AIによるネットワーク運用

通信事業者が高品質な通信サービスを顧客に提供するためには、高品質なネットワーク運用が求められる。

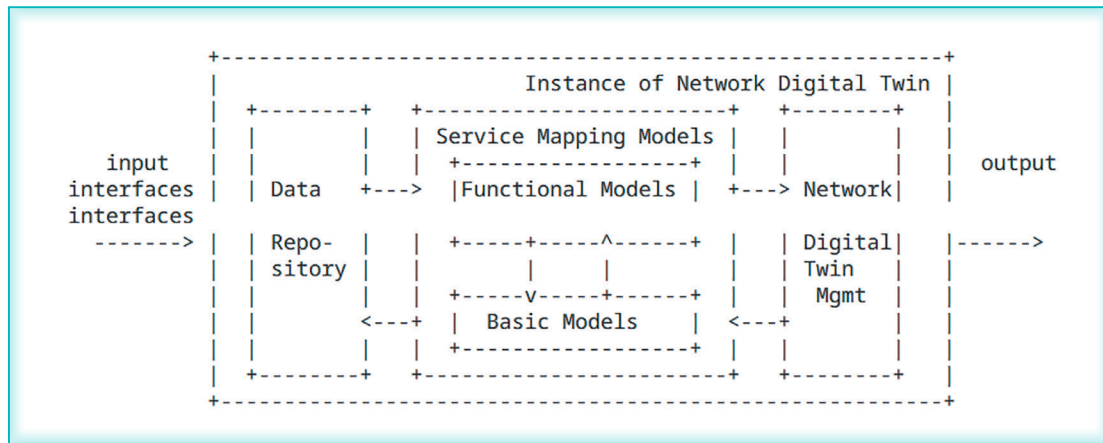
そのために、AI技術をネットワーク運用へ適用する試みが広く進められており^{(1)・(9)~(11)}、一部の運用手順（例：障害検知）に対する商用化も開始されている⁽¹²⁾。

特に、ネットワーク上で発生するネットワーク障害への高精度かつ迅速な対応のために、AI技術の適用が検討されている。ネットワーク障害への対応は①障害検知⁽⁹⁾、②原因分析⁽¹¹⁾、③復旧⁽¹⁰⁾という三つのステップで一般的には構築されており、それぞれのステップに対して機械学習、深層学習、強化学習等のAI技術の適用が検討されている。

2.2 ネットワークデジタルツインとAIモデル構築

近年、通信事業者が運用する商用ネットワークを、仮想環境上で仮想的に構築するNDTに関する研究開発や、標準化の活動が盛んに進められている^{(2)・(3)・(13)~(15)}。NDTは、図2で示されるように、商用ネットワークの運用データをインプットとして構築され、仮想環境上で構築されたNDT上で、商用ネットワーク上では実施が困難なオペレーション（例：障害試験、品質分析、作業手順確認）が可能となる。

特に、NDTのユースケースの一つとして、AIモデルの構築が挙げられる^{(2)・(3)}。ネットワーク運用へのAI技術の適用が注目される一方で、そのAIモデルを構築するプラットフォームは欠かせない要素であり、NDT上でAIモデル構築に必要となる各

図2 NDT アーキテクチャの標準例⁽²⁾

操作（例：擬似障害の発生，統計情報の取得）を行うことで，商用ネットワークに影響を与えずに AI モデルの構築が可能となる。

3 関連研究：5G モバイルネットワーク

3.1 5G モバイルネットワーク

第5世代移動通信システム（5G）⁽¹⁶⁾ は，図3で表されるように，全国に分散された基地局ネットワーク（RAN: Radio Access Network）と，RANが接続を行うモバイルコア（Mobile Core）で構築される。

RANにおいては，5Gを利用するUser Equipment（UE）と電波による通信を行うRadio Unit（RU），RUと送受信を行い無線信号の信号処理を行うDistributed Unit（DU），DUとRUの制御を行いモバイルコアと接続するCentral Unit（CU）で構成される⁽¹⁷⁾。

次に，モバイルコア，特に5Gのモバイルコアである5G Core Network（5GC）は，UEの接続要求，認証，モビリティ管理といったControl-plane

（C-Plane）に関する処理と，実際にUEが外部ネットワーク（例：インターネット，Multi-access Edge Computing(MEC)）と通信するIPパケット／Ethernetフレームの転送処理，課金管理といったUser-plane（U-Plane）に関する処理を行う⁽¹⁶⁾。更に，5Gから採用されたService-Based Architecture（SBA）⁽¹⁶⁾では，クラウドネイティブ⁽¹⁸⁾なアーキテクチャを目指し，5GCにおいて必要となる役割ごとにNetwork Function（NF）として分割されており，C-Plane/U-Planeに関して必要となる各処理（例：移動管理，セッション管理，認証，加入者情報管理）ごとにNFが構築され，NF間をAPIで接続することで，スケーラブルな構成を採用している。

3.2 5GCにおけるスケールアウトの取組み

特に，全国のUEからの接続要求が集中する5GCにおいては，大規模な処理能力が求められる。一般的に，計算基盤において大規模な処理能力を持つために，単体の計算インスタンスの規模を増大するスケールアップと，同一の計算インスタンスを並列で配置するスケールアウトが採用される⁽¹⁹⁾が，

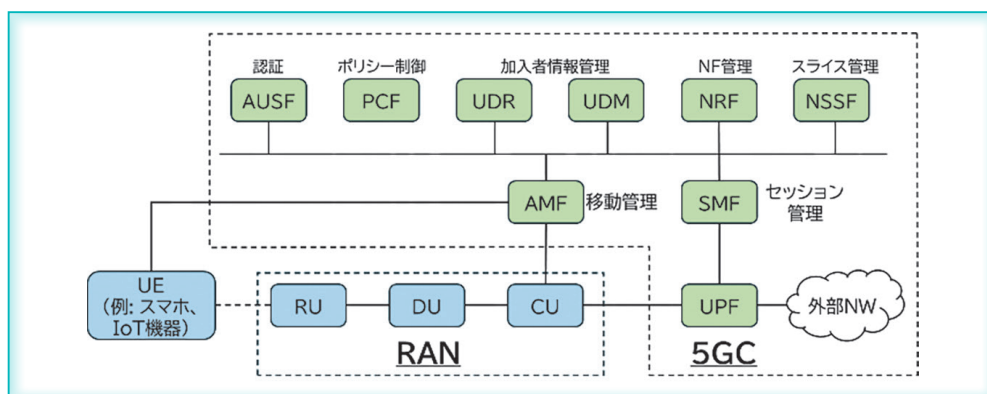


図3 5G モバイルコア構成

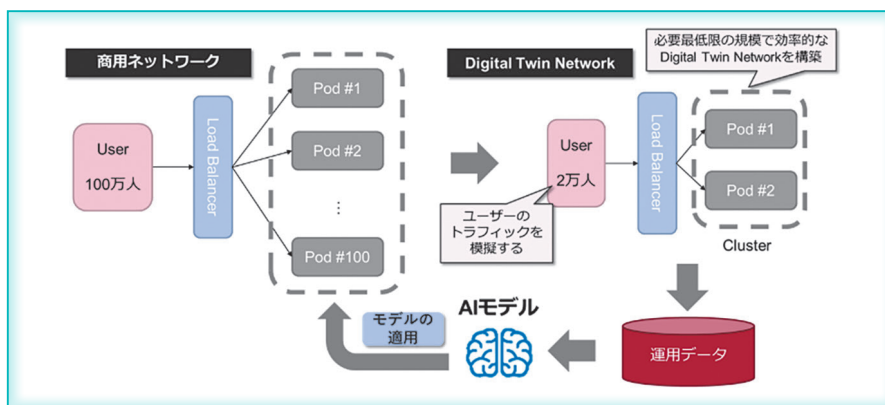


図4 スケールアウト構成を用いた NDT

5GC においては近年スケールアウトの適用に関する検討が広く進められている^{(4)~(7)}。

5GC においてスケールアウトの採用に関する検討が加速している要因としては、5G から採用された Service-Based Architecture (SBA) にある⁽¹⁶⁾。SBA においては、5GC 内部 NF を必要な性能に即して、仮想 VM / コンテナとして構築することで、需要に合わせた柔軟な 5GC の構築が可能となる。

近年、コンテナ化アプリケーションの大規模運用において Kubernetes⁽²⁰⁾ が広く用いられている。Kubernetes は、所望の状態（例：レプリカ数や計算資源量）を宣言し、コントローラがクラスタの状態を継続的に監視・調整して、その状態を維持するオーケストレーション基盤である。スケールアウトは、レプリカ数の動的調整（更新制御や自動スケールリング機構）によって達成され、外部からの到達点は Kubernetes Service が複数の実行単位を単一の仮想エンドポイントに集約することで到達可能となる。Kubernetes Service はクラスタ内の名前解決機構と連携して宛先を解決し、外部からのトラヒックを各 Pod へ分配する。これにより、例えば 5GC における各 NF 内において Pod を並列に構成した場合でも、一貫したエンドポイントを保ちながら、需要に応じて処理能力をスケラブルに拡張することが可能となる。

4 スケールアウト構成のネットワークデジタルツインを用いた AI モデル構築

4.1 提案手法概要

ネットワーク運用に用いられる AI モデルを訓練するためには、①ネットワーク構成と②トラヒックの二つの要素の検討が必要となる。初めに、①のネットワーク構成とは、ネットワークを構成する各ノ

ードの CPU/Memory の規模、接続されたネットワーク帯域量といったインフラに関する情報である。また、②のトラヒックとは、そのネットワークを利用するユーザが、ネットワークに対してリクエスト（例：Web 閲覧、5GC への接続・ハンドオーバ）を実施することで発生する IP パケットの集合である。

商用ネットワークに対して適用可能な高精度な AI モデルを訓練するためには、上記 2 要素に関して商用ネットワークと同じ環境を用意し、障害試験等の必要となる各種試験を実施し、ネットワークから得られた運用データ（例：時系列データ、Syslog）を収集し、そのデータを入力とした AI モデルの訓練を行う必要がある。

しかしながら、2. で述べたように、商用ネットワークと完全に等しいネットワーク構成・トラヒックの構築には、商用ネットワークと完全に同規模の設備を持つ必要があり、設備コストが倍増してしまう課題が存在する。そこで本研究では、スケールアウトの技術を活用し、「スケールアウト構成の最小構成要素」のみを NDT 上で構築することで、必要最低限の NDT で商用に適用可能な精度の AI モデルの訓練を目指す。

4.2 提案手法詳細

ここで、スケールアウト構成を用いた NDT 環境の構成図を図 4 に示す。スケールアウト構成では、均一なネットワークインスタンスを並列に多数並べ、多数のユーザからのリクエストをその並列化されたネットワークインスタンスに、均一にロードバランスさせるアーキテクチャとなる⁽¹⁹⁾。そのため、例えば商用ネットワークが 100 個のコンテナから構成されており、それらを用いて 100 万人のユーザを処理している場合、NDT 環境においてはコンテナ 1 個に対し $100 \text{ 万} \div 100 \text{ コンテナ} = 1 \text{ 万人分}$

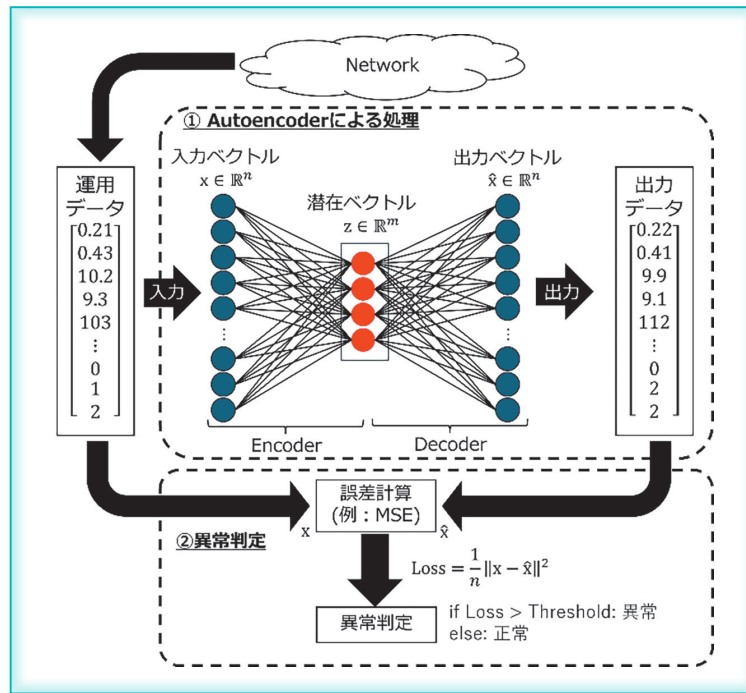


図5 Autoencoder による障害検知

のユーザの模擬トラフィックをかけることで、商用ネットワークと同等の環境を再現することができる。このようにして、単一のネットワークインスタンスに収容されるユーザ規模を同等にするように、トラフィックを調整することで、最低限の規模で、大規模な商用ネットワークと同等の環境が構築可能となる。

その後、NDT 環境にて運用データの収集を行い、インスタンス単位で AI モデルの学習・構築を行う。各インスタンスの規模は商用ネットワークと NDT 環境で同等のため、作成した AI モデルは商用ネットワークの各インスタンスに反映することが可能となる。このように、スケールアウト構成を活用することで、大規模な商用環境の全てを再現せずに、小規模な検証環境で AI モデルを構築することが可能となる。

5 Autoencoder による障害検知

5.1 Autoencoder 概要

2. で述べたように、AI によるネットワーク運用では様々なユースケースが検討されているが、特に本稿では、「障害検知」に関する観点で以降の評価を実施する。評価に当たり、ニューラルネットワークを用いた次元圧縮・復号のアルゴリズムである Autoencoder⁽²¹⁾ を用いて評価を行う。

Autoencoder は図5で示されるように、

Encoder 層と Decoder 層によって表現され、入力ベクトル $x \in \mathbb{R}^n$ を Encoder 層へ入力し、低次元の潜在ベクトルである $z \in \mathbb{R}^m$ に次元圧縮する。 z と x の関係は以下式で表現される。

$$z = f_{enc}(x) = \sigma(W_{enc}x + b_{enc})$$

ただし、 $W_{enc} \in \mathbb{R}^{m \times n}$ は Encoder 層における重み行列、 $b_{enc} \in \mathbb{R}^m$ は Encoder 層におけるバイアスペクトル、 $\sigma(\cdot)$ は ReLU や tanh などの活性化関数である。このようにして、 n 次元の入力ベクトルを m 次元の潜在ベクトル z に次元圧縮することができる。

次に、Decoder 層では、逆に m 次元の潜在ベクトルを n 次元のベクトルである $\hat{x}$ に復元する。 z と $\hat{x}$ の関係は以下の式で表現される。

$$\hat{x} = f_{dec}(z) = \sigma(W_{dec}z + b_{dec})$$

ただし、 $W_{dec} \in \mathbb{R}^{n \times m}$ は Decoder 層における重み行列、 $b_{dec} \in \mathbb{R}^n$ は Decoder 層におけるバイアスペクトルである。このようにして、 m 次元の潜在ベクトルを n 次元のベクトル $\hat{x}$ に復元することができる。

Autoencoder 学習の目的は入力 (x) と出力 ($\hat{x}$) の差を最小化することであり、学習ステップにおいては、損失関数として、以下のような x と $\hat{x}$ の平均二乗誤差 (MSE) などが用いられる。

$$\frac{1}{n} \|x - \hat{x}\|^2 = \frac{1}{n} \sum_{i=0}^n (x_i - \hat{x}_i)^2$$

上記のようなモデル、損失関数を基に、大量の学

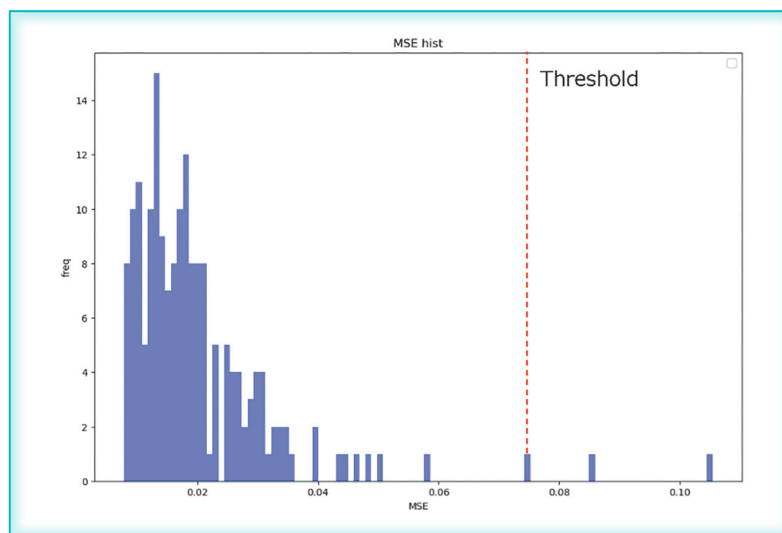


図6 しきい値決定方法の例

習データを用いて、誤差逆伝搬法や勾配降下法などを用いて学習が行われ、各パラメータの最適値が決定され、Autoencoderの学習が完了する。

次に、学習が完了したAutoencoderモデルを用いた異常判定方法について記述する。上記で記したように、AutoencoderのEncoder/Decoder層を通すことで、出力ベクトルとしては入力ベクトルと近いものが出力されることとなる。しかしながら、学習に用いたデータと異なる傾向の入力ベクトルを入力した場合、その傾向のベクトルが学習されていないため、入力ベクトルと出力ベクトルの誤差が大きくなる。この特性を用いることでAutoencoderによる異常判定を実現することができる。

例えば、入力ベクトル x と出力ベクトル $\hat{x}$ の平均二乗誤差(MSE)を用いた異常判定を実施する場合、図6に示されるように、学習時のデータセットにおける x と $\hat{x}$ のMSEの集合値からしきい値を決定する。図6の例においては、MSEの集合における95パーセンタイル値をしきい値としている。(例えば、以下で記述する評価環境においては、UDR1個の環境におけるAIモデルのしきい値は、0.008であった。)本しきい値の値は異常検知の精度に影響があり、例えばしきい値を下げることで、「見逃し数」を低下させるが「誤検知数」が上昇する恐れがあり、反対にしきい値を上げることで、「誤検知」を低下させるが、「見逃し数」が上昇する恐れがあり、各データセットにおいて最適なしきい値を設定する必要がある。

5.2 AIモデルの評価指標

次に、一般的なAIモデルの精度を評価するための数値指標として、適合率(Precision)、再現率

(Recall)、F1スコアについて説明する。全体のデータセット数の中でAIモデルの結果としては、以下の四つに分類される：TP(True Positive, AIモデルが「正」と予測し、実際も正だった件数)、FP(False Positive, AIモデルが「正」と予測したが、実際は「負」だった件数、誤検出)、FN(False Negative, AIモデルが「負」と予測したが、実際は正だった件数、見逃した数)、TN(True Negative, AIモデルが「負」と予測し、実際も「負」だった件数)。

その上で、適合率(Precision)は、AIモデルが「正」と判定したもののうち実際に正であった割合であり、 $Precision = TP / (TP + FP)$ と定義される。これは誤検出(FP)をどれだけ抑えられているかを測る指標であり、誤検出を減らすことを重視するユースケースで有用な評価指標である。一方、Precisionを高めるために判定を厳しくすると見逃し(FN)が増え、Recallとのトレードオフが生じる。

一方で、再現率(Recall)は、実際に正である対象のうちAIモデルが正と判定できた割合で、 $Recall = TP / (TP + FN)$ と定義される。これは見逃し(FN)をどれだけ抑えられているかを表し、見逃しを減らすことが重要な場面で重視される。ただし、判定を緩くしてRecallを上げると誤検出(FP)が増え、Precisionが低下し得る。

最後に、F1スコアは適合率と再現率の調和平均であり、 $F1 = 2 \times Precision \times Recall / (Precision + Recall)$ と定義される。調和平均であるため、PrecisionかRecallのどちらか一方が低いと全体も低く評価され、誤検出(FP)と見逃し(FN)の双方をバランス良く表現した評価指標となる。F1ス

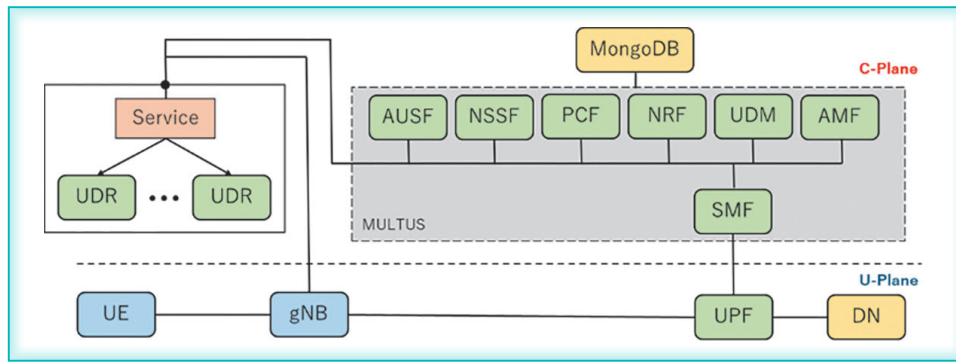


図7 モバイルコアのスケールアウト構成

コアの最大値は1であり、1に近い方が良いAIモデルであると言える。

6 評価環境

6.1 評価対象となるスケールアウト5GCの構築

初めに、本提案手法の初期評価のために、スケールアウトされた5GC環境を構築する必要がある。近年、free5GC⁽⁸⁾といったOSSな5GCソフトウェアが盛んに開発されており、本研究では、スケールアウト構成の基盤としてコンテナオーケストレーションツールであるKubernetes⁽²⁰⁾を用いて、free5GCの環境を構築する。また、free5GCにおいてユーザーデータの管理を行うUDR（Unified Data Repository）コンテナをスケールアウトした環境を構築する。

より詳細には、まずはUDRのNetwork Function（NF）を並列に並べる構成を検討する。free5GCの実装では、各コンテナインタフェースの接続先情報は構成ファイル上で静的に定義されている。したがって、NFを複数構築し並列に並べた

場合、インタフェースのIPアドレスが重複してしまう。それを回避するためにはDHCPサーバや負荷分散のためのDNSサーバの設置が必要となり、ネットワーク構成が複雑となる。

そのため、Kubernetes Serviceを用いたスケールアウト構成を構築する。Kubernetes Serviceは複数のPodへの接続ポリシーを定義し、Pod集合に対してアクセスするためのエンドポイントを提供する。各Serviceには一意のIPアドレスが割り当てられ、配下のPod群に対して負荷分散と名前解決が行えるため、Serviceを用いて並列にNFを構築することで、スケールアウトなNF構成を実現することができる（図7）。

6.2 評価環境の構成

評価環境の全体構成を図8に示す。評価環境として、図7に示したスケールアウト構成となる5GC構成をKubernetes上でfree5GCを用いて構築する。その上で、UDRの配置数を変化させることで、規模の異なる環境におけるAIモデルの訓練・評価が可能となる。

また、インフラ基盤からのデータ収集にはeBPF

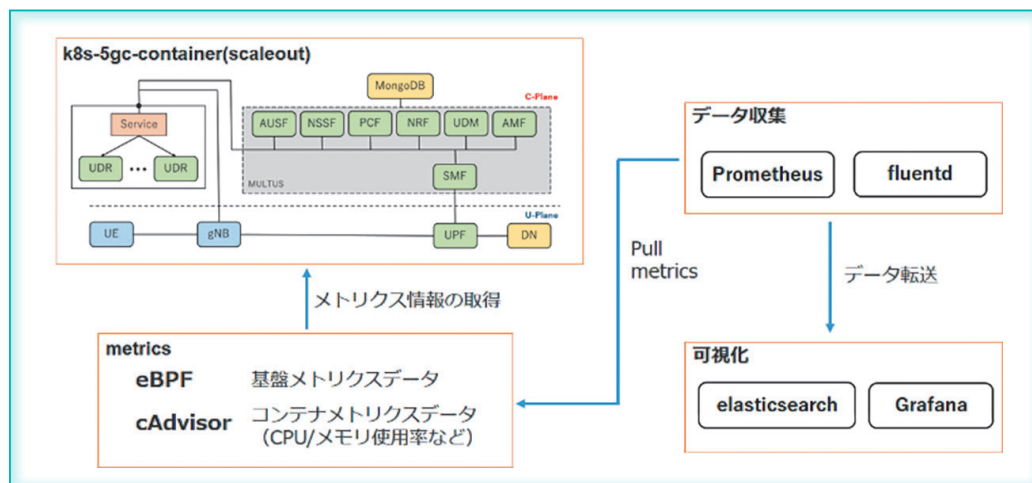


図8 評価環境

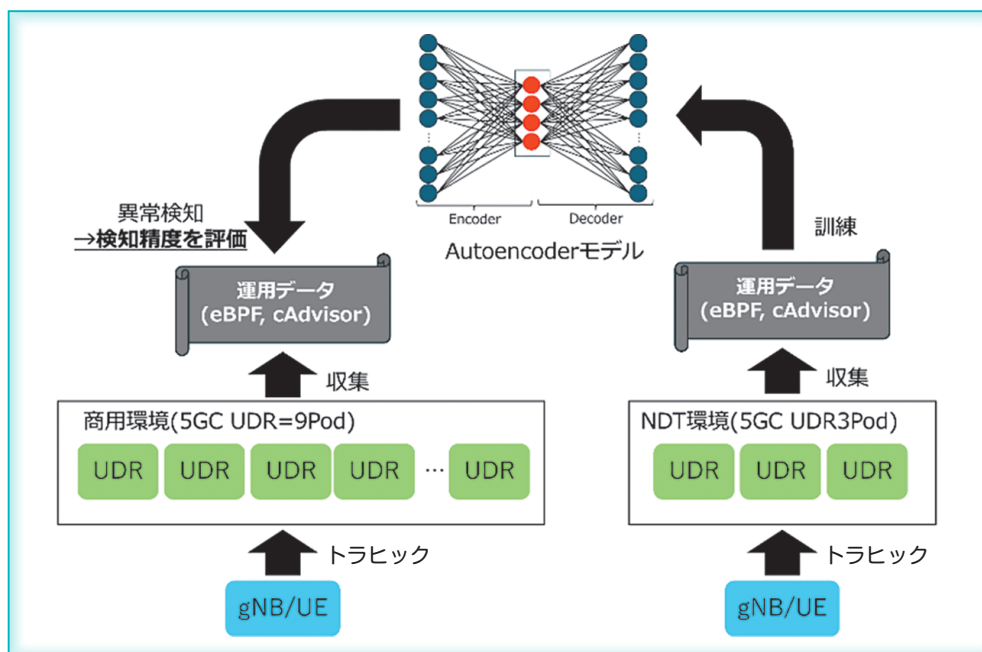


図9 異なる環境下の障害検知精度評価方法

(extended Berkeley Packet Filter)⁽²²⁾ と cAdvisor⁽²³⁾ を利用している。eBPF は Linux kernel 3.15 以降のモニタリング機能であり、各プロセスにおける Kernel 空間上で発生するイベント(例: TCP 再送発生数)に関する詳細モニタリングデータが取得できる。cAdvisor は Kubernetes 環境におけるオープンソースモニタリングツールであり、実行中の各コンテナの基本的なメトリクス(例: CPU 使用率、トラフィック量)の収集が可能となる。そのほか、メトリクス情報の取得には OSS である Prometheus⁽²⁴⁾ と Fluentd⁽²⁵⁾ を利用し、メトリクス情報の可視化には Elasticsearch⁽²⁶⁾ と Grafana⁽²⁷⁾ を利用した。

6.3 評価方法

スケールアウト環境における AI モデル構築手法の初期評価を行うために、上述のスケールアウト構成の 5GC において、規模の異なる環境に対して AI モデルを適用した際の精度評価を行う。本評価においてはネットワーク障害検知のために Autoencoder を用い、その障害検知精度に関して評価を実施する。本評価では、Autoencoder の各パラメータとして、入出力ベクトルはいずれも 129 次元であり、Encoder 側に 8 ユニットの隠れ層を 1 層持ち、その出力(潜在ベクトル)は 8 次元であり、Decoder 側において 8 次元から 129 次元へ再構成している。最適化手法としては Adam を利用し、損失は MSE、学習はエポック 200、バッチサイズ 32 で実施している。

図9は、異なるスケールアウト構成間における障害検知精度の評価方法を表している。図9に示されるように、はじめに、NFの規模が異なるスケールアウト構成の5GCを用意し、各環境でAIモデルの訓練を行う。ここで異なる規模の環境はNFのPod数を変化させることで構築するものとし、NFのPod数がそれぞれ1, 3, 5, 7, 9個の合計5環境を用意する。図9の例では、NDT環境模擬として、3個のUDRで構成された環境となる。

その後、各環境で訓練した Autoencoder モデルをほかの環境に適用し、ネットワーク障害が発生した場合の障害検知精度を評価する。図9の例では、商用環境模擬として、9個のUDRで構成された環境となる。

また、今回の評価においては、5GC内で発生するネットワーク障害シナリオとしては、Kubernetes ネットワーク内においてネットワーク遅延の増大を想定し、その結果、各環境において各NF間のTCP通信の遅延時間が増大するものとする。

ここで、ネットワークに遅延をかける手段として Linux TC コマンド (traffic control コマンド) を利用した。正常時の遅延時間として、各NF間の遅延時間を 10 ms として設定し、ネットワーク障害時においては、この遅延時間を 50 ms に増加させることで、遅延時間を増大させている。

また、データセットの取得方法として、各五つの環境において、ネットワーク遅延の障害発生を想定したテストシナリオとして、計2時間の擬似呼の

表 1 ネットワーク障害の検知精度評価結果

一列目：NDT 環境として用いた構成の Pod 数（訓練）
 一行目：商用環境として用いた構成の Pod 数（評価）

	1	3	5	7	9
1	0.667	0.817	0.8282	0.816	0.829
3	0.809	0.819	0.842	0.828	0.835
5	0.807	0.820	0.857	0.824	0.840
7	0.810	0.820	0.840	0.820	0.830
9	0.808	0.820	0.852	0.823	0.842

生成を行い、その間 UE が attach/detach を繰り返す動作を実行させている。この 2 時間のうち、前半 1 時間を正常な状態（遅延時間 10 ms）とし、その後遅延時間を 50 ms に増大させることで後半 1 時間をネットワーク遅延障害が発生している期間とした。その後正常データ 1 時間のうち、初めの 30 分を学習データとして利用し、残りの正常期間 30 分＋障害発生期間 1 時間半を検証データとして利用した。

7 評価結果

7.1 異なるネットワーク規模間の障害検知精度比較

前章で構築した評価環境にて AI モデルを訓練し、Pod 数が異なるほかの環境へその AI モデルを適用し、各環境のネットワーク障害の検知精度に関して F1 Score を算出することで精度評価を実施した。各環境に対して算出した F1 Score の結果を表 1 に示す。ちなみに、F1 Score は、適合率（Precision）と再現率（Recall）の調和平均であり、分類問題における総合的なモデル性能評価指標となり、0～1

の値を取り 1 に近い方が良い結果となる。

表 1 の一列目は図 9 における NDT 環境となり、Autoencoder モデルを訓練するために利用した Pod 数を示しており、一行目は図 9 における商用環境となり、訓練した Autoencoder モデルを適用し、ネットワーク障害が検知できるか評価するために利用した Pod 数を示す。例えば、図 9 の環境では、Autoencoder モデルを訓練するための NDT 環境としては UDR の Pod が 3 個、その Autoencoder モデルを適用する商用環境として UDR の Pod が 9 個となるため、その場合の F1 Score としては表 1 でハイライトされている 0.835 となる。

評価結果から、Autoencoder モデルを訓練する NDT 環境の Pod 数が 1 個と少ない場合でも、Pod 数が 3 個と NDT 環境よりも大きい規模の商用環境に対して高い F1 Score が得られている。このように、NDT 環境の Pod 数にかかわらず、規模の異なる商用環境でネットワーク障害の検知が可能となると言える。ただし、Pod 数 1 個同士での F1 Score (0.667) がほかと比べて低くなっている要因としては、Pod 数 1 個の場合は、全ての通信が同一 Pod に流入し、トラヒックの絶対量が増加し、障害発生時の変化が相対的に小さくなってしまったことが考えられる。

また、評価環境に AI モデルを適用した際の時系列の異常検知結果を図 10 に示す。図 10 では、NDT 環境の Pod 数が 1 個で Autoencoder モデルを構築し、Pod 数を 3 個にスケールアウトした構成の商用環境に Autoencoder モデル適用している。図 10 にて実施した評価試験では、時系列データの後半部分（図 10 の 22:00 以降）にネットワーク障

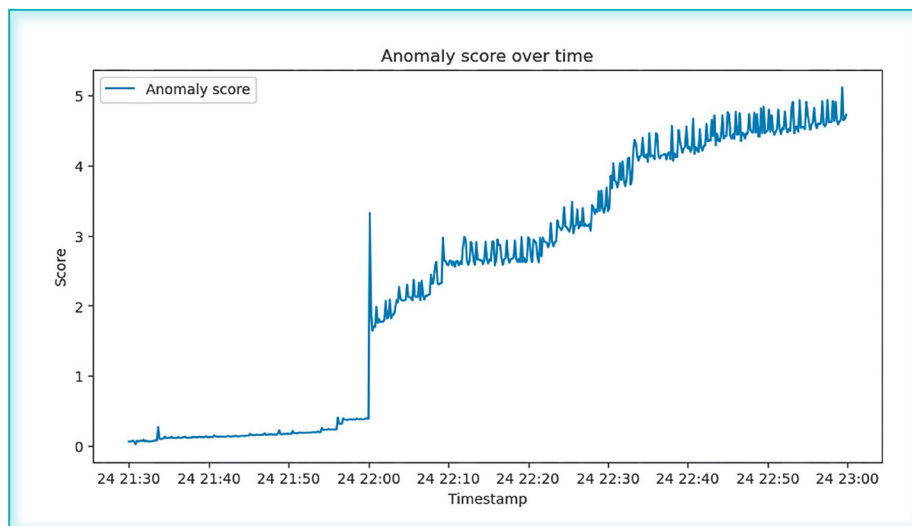


図 10 ネットワーク障害発生時の異常スコア変動

表2 Pod 数 9 個環境における各 Pod ごとの障害検知精度 (F1 Score)

	Pod#1	Pod#2	Pod#3	Pod#4	Pod#5	Pod#6	Pod#7	Pod#8	Pod#9
1	0.808	0.840	0.892	0.802	0.820	0.844	0.837	0.809	0.820
3	0.829	0.827	0.837	0.840	0.815	0.838	0.841	0.822	0.839
5	0.817	0.827	0.830	0.816	0.821	0.819	0.818	0.818	0.828
7	0.818	0.838	0.857	0.818	0.819	0.836	0.846	0.829	0.845

害（通信遅延増大）を発生させており、ネットワーク障害が発生したタイミングを境に異常スコアの上昇が見られ、NDT 環境で構築した Autoencoder モデルの動作の様子が確認できる。障害発生後、異常スコアが時間経過とともに上昇している要因としては、5GC 内部にて障害発生に伴い、UE のタイムアウトや再接続などのイベントが徐々に発生し、今回の対象である UDR への接続数が通常時に比べて徐々に増加することが原因であると考えられる。また、障害発生直後に急激に異常スコアが上昇している要因としては、その時点において、急激に通信遅延が増大（10 ms → 50 ms）し、通信遅延量などのメトリクスが急激に変化したためと考えられる。

7.2 スケールアウト構成における負荷分散に関する考察

次に、スケールアウト構成において、並列に構築された Pod ごとのネットワーク障害検知精度の評価を実施する。本評価においては、Pod 数が 1, 3, 5, 7 個の各 NDT 環境で訓練した Autoencoder モデルを、Pod 数が 9 個の商用環境における各 9 個の UDR Pod に対して適用した場合の各 Pod におけるネットワーク障害の検知精度に関して評価を実施しており、表 2 がその結果となる。

表 2 から各 Pod におけるネットワーク障害の検知精度にある程度の差が見られるが、全ての Pod において高い精度でネットワーク障害を検知できていることが分かる。Pod 間でネットワーク障害の検知精度に差がある理由としては、スケールアウト構成において UE からリクエストする接続先 Pod が完全に分散されておらず、ばらつきが存在することが原因と考えられる。本構成のように、Kubernetes Service を利用したスケールアウト構成を用いる場合、Service は Pod 集合に対してアクセスするためのエンドポイントを提供し、そのエンドポイントへ一意の IP アドレスが割り当てられるため、配下の Pod 群に対して負荷分散と名前解決が可能となる。仮に、Kubernetes における各 Pod への負荷分散にばらつきが存在する場合、Pod に流入するリクエスト数が不均一となり、各 Pod 上

で動作するプロセスの振る舞いやリソース（CPU/Memory/Network）の消費の傾向に違いが出る。その結果として、eBPF と cAdvisor で得られるデータの傾向に Pod 間差分が生じることにより、障害検知の精度に差が生じたと考えられる。このことにより、スケールアウト構成における負荷分散が均一に実施されることが、本手法が有効となる前提であると言える。

8 おわりに

本稿では、スケールアウトされた 5GC 環境においてネットワークデジタルツインの技術を活用し、ネットワーク運用のための AI モデル構築手法を提案した。本手法により、大規模な商用環境を最小構成の設備で再現可能となり、商用ネットワーク上では実現困難な、各ネットワーク障害の試験をネットワークデジタルツイン環境において実施が可能となる。また、試験を通じて得られる運用データセットを基に AI モデルを構築することで、作成した AI モデルを商用ネットワーク環境へ反映することが可能となることを示した。今後は、本手法で必要となるネットワークデジタルツインの動的な構築手法を検討し、完全に自律的なネットワーク運用を目指す。

文献

- (1) P. Notaro, J. Cardoso, and M. Gerndt, "A survey of aiops methods for failure management," ACM Trans. Intell. Syst. Technol., vol. 12, no. 6, pp. 1-45, Nov. 2021.
- (2) C. Zhou, H. Yang, X. Duan, D. Lopez, A. Pastor, Q. Wu, M. Boucadair, and C. Jacquenet, "Digital twin network: concepts and reference architecture," Internet-Draft draft-irtf-nmrg-networkdigital-twin-arch-04, Internet Engineering Task Force, Oct. 2023.
- (3) ITU-T, "Architectural framework for artificial intelligence-based network automation for resource and fault management in future networks including IMT-2020," Recommendation Y.3177, Feb. 2021.
- (4) V. Nagendra, A. Bhattacharya, A. Gandhi, and S. R. Das. Mmlite, "A scalable and resource

- efficient control plane for next generation cellular packet core,” Proc. 2019 ACM Symposium on SDN Research, SOSR, pp. 69–83, April 2019.
- (5) D. Kumar, S. Chakrabarti, A. S. Rajan, and J. Huang, “Scaling telecom core network functions in public cloud infrastructure,” IEEE Int. Conf. Cloud Comput. Technol. and Sci. (CloudCom), pp. 9–16, Dec. 2020.
 - (6) C. Rotter and T. Van Do, “A queueing model for threshold-based scaling of upf instances in 5g core,” IEEE Access, vol. 9, pp. 81443–81453, June 2021.
 - (7) D. Harutyunyan, R. Behraves, and N. S. Kriještorac, “Cost-efficient placement and scaling of 5g core network and mec-enabled application vnfs,” 2021 IFIP/IEEE Int. Symposium Integrated Netw. Management(IM), pp. 241–249, May 2021.
 - (8) free5GC. <https://www.free5gc.org/>
 - (9) J. Kawasaki, D. Koyama, T. Miyasaka, and T. Otani, “Failure prediction in cloud native 5g core with ebpf-based observability,” 2023 IEEE 97th Vehicular Technol. Conf. (VTC2023-Spring), pp. 1–6, June 2023.
 - (10) T. Miyamoto, G. Mori, Y. Suzuki, and T. Otani, “Network topology-traceable fault recovery framework with reinforcement learning,” Adv. Inf. Netw. and Appl., pp. 393–402, Cham, 2021, Springer International Publishing.
 - (11) J. Kawasaki, G. Mouri, and Y. Suzuki, “Comparative analysis of network fault classification using machine learning,” 2020 IEEE/IFIP Netw. Oper. Manag. Symposium, pp. 1–6, April 2020.
 - (12) KDDI, “ニュースリリース AIを活用した障害検知システムの運用を開始,” Jan. 2024. <https://news.kddi.com/kddi/corporate/newsrelease/2024/01/25/7194.html>
 - (13) Y. Wu, K. Zhang, and Y. Zhang, “Digital twin networks: A survey,” IEEE Internet of Things Journal, vol. 8, no. 18, pp. 13789–13804, Sept. 2021.
 - (14) H. X. Nguyen, R. Trestian, D. To, and M. Tatipamula, “Digital twin for 5G and beyond. IEEE Commun. Mag., vol. 59, no. 2, pp. 10–15, March 2021.
 - (15) H. Hong, Q. Wu, F. Dong, W. Song, R. Sun, T. Han, C. Zhou, and H. Yang, “Netgraph: An intelligent operated digital twin platform for data center networks,” Proc. ACM SIGCOMM 2021 Workshop Netw.-Appl. Integration, pp. 26–32, Aug. 2021. Association for Computing Machinery.
 - (16) 3GPP, “System architecture for the 5G System (5GS),” Technical Specification (TS) 23.501, 3rd Generation Partnership Project (3GPP), 12 2023. Version 18.4.0.
 - (17) 3GPP, “NG-RAN; Architecture description. Technical Specification (TS) 38.401, 3rd Generation Partnership Project (3GPP), 01 2024. Version 18.0.0.

- (18) CNCF Cloud Native Definition v1.0. <https://github.com/cncf/toc/blob/main/DEFINITION.md>
- (19) K. Hwang, Y. Shi, and X. Bai, “Scale-out vs. scale-up techniques for cloud performance and productivity,” IEEE 6th International Conf. Cloud Comput. Technol. and Sci., pp. 763–768, Dec. 2014.
- (20) Kubernetes. <https://kubernetes.io/>
- (21) G. E. Hinton and R. R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” Science 313, pp. 504–507, Aug. 2006.
- (22) eBPF. <https://ebpf.io/>
- (23) cAdvisor. <https://github.com/google/cadvisor>
- (24) Prometheus. <https://prometheus.io/>
- (25) Fluentd. <https://www.fluentd.org/>
- (26) Elasticsearch. <https://www.elastic.co/>
- (27) Grafana. <https://grafana.com/>

(2025年8月9日受付, 11月10日再受付)

小山大輝

2017中大・院・修士課程了。同年KDDI入社。2021～KDDI総合研究所でネットワーク運用自動化、ネットワーク障害の予兆検知に関する研究開発に従事。2024に帰任後、通信品質改善に向けたビッグデータ解析業務に従事。



桜庭皆人 (正員)

2014科学大・院・修士課程了。同年KDDI入社。モバイルIP網のバックホール収容ネットワーク開発に従事。2022KDDI総合研究所に出向し、現在まで自律型ネットワークにおけるオーケストレーションに関する研究に従事。



河崎純一 (正員)

2011慶大・院・修士課程了。同年KDDI入社。運用部門においてグローバルネットワークのオペレーション業務に従事。現在、KDDI総合研究所においてAIを活用した運用自動化として障害検知・原因解析等の研究及び実用化に従事。



宮坂拓也 (正員：シニア会員)

2011東大・院・修士課程了。同年KDDI入社。国内バックボーンネットワークの開発業務に従事。2018KDDI総合研究所に出向し、現在までネットワーク運用に関する研究開発や標準化に従事。



トラフィックデータの分析による 情報通信ネットワークの異常検知

鎌村星平 Shohei Kamamura 成蹊大学

1 はじめに

Society 5.0 と呼ばれる、サイバー空間と現実空間が高度に融合した超スマート社会の実現に向けて、情報通信ネットワークは社会基盤としてこれまで以上に重要な役割を担っている。例えば、IoT 機器や AI を活用した多様なサービスやアプリケーションは、一般的には互いにネットワークを介して連携して動作する。よって、これらを結ぶ情報通信ネットワークの安定性や信頼性の確保が重要となる。

安定した情報通信ネットワークを構築して運用するためには、トラフィック情報が必要不可欠である。トラフィックとはインターネットなどの情報通信ネットワーク上で送受信されるデータの量や頻度、そのパターンを指す言葉である(図 1)。通信事業者はこのトラフィック情報のトレンドを分析し、過不足のない適切なネットワーク設備設計を行う⁽¹⁾、また、トラフィックの日常からの変化を分析することで、サイバー攻撃や設備故障といったリアルタイムでの異常検知も可能となる⁽²⁾。更に、トラフィックから人間の行動や思考を抽出することで、Society 5.0 に直接結び付く新たな価値の創造も期待されている⁽³⁾。

本稿では、トラフィック情報を活用した事例として、ト

ラフィックの変化を数学的に解析することで情報通信ネットワークの異常検知を行う方法を解説する。以降、まず **2.**でトラフィックの観測方法を体系的に解説する。**3.**では時系列データとして観測されたトラフィック情報の自己相関分析によって異常検知を行う方法を解説する。**4.**では複数のトラフィック情報の相関を空間的に分析し、異常箇所を推定する方法を解説する。最後に**5.**で総括し、今後の展望を述べる。

2 トラフィックの観測方法

トラフィックの観測方法を分類したものを図 2 に示す。トラフィックの観測方法は主に従来から使われる SNMP (Simple Network Management Protocol) やネットワークテレメトリー⁽⁴⁾といったルータが保有するパケットカウンタ値を取得する方法、すなわちリンクトラフィックを観測する方法と、「通信フロー」と呼ばれるパケットの集合体を、サンプリングや確率的データ構造を用いて観測する方法とに分類される。ここで通信フローとは、送信元 IP アドレス、宛先 IP アドレス、送信元ポート番号、宛先ポート番号、プロトコルタイプの五つの識別子 (5-tuple) を用いて識別されるパケットの集

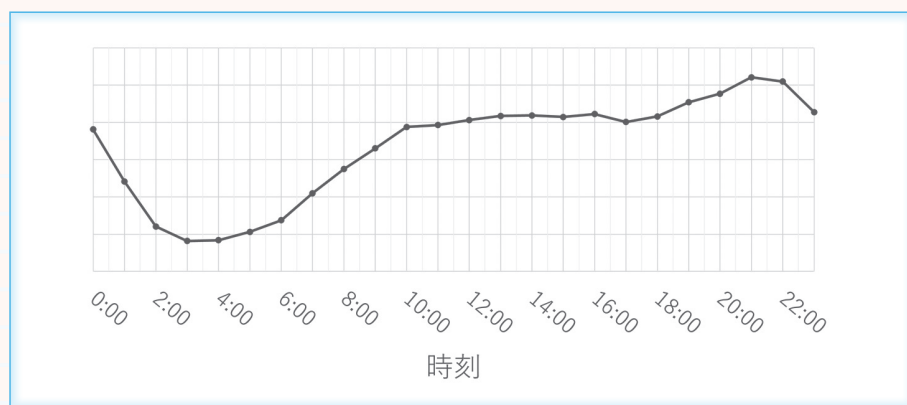


図 1 時系列トラフィック (昼夜変動) の例

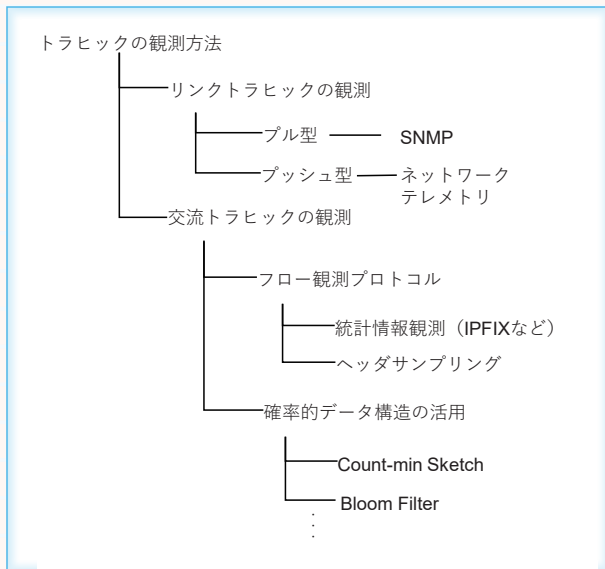


図2 トラヒック観測方法の分類

合体である。リンクトラヒックは観測が容易であるが、その量はリンクを通過する合算値として粗い粒度で情報が得られる。一方で通信フローは、通信フロー観測プロトコル⁽⁵⁾をルータに実装する必要があるが、ルータ上で一定の負荷が発生するが、ルータ間やユーザ間といった特定の対地間のトラヒック発生状況を精緻に把握することが可能となる。これら特定の対地間のトラヒックを

「交流トラヒック」と呼ぶ。リンクトラヒックと交流トラヒックを図3に示す。リンクトラヒックの量は、そのリンクを通過する交流トラヒックの和となる線形の関係がある。

次に、通信フロー観測プロトコルを用いてトラヒックを観測する方法を説明する(図4)。まず、(1) IP ルータ上で一定のルール及びサンプリング周期で IP パケットのヘッダ情報を観測する。次に、(2) IP パケットのヘッダ情報に基づいて定義される通信フローごとの出現量をカウントする。最後に、(3) サンプリングレートに基づきサンプリング前のトラヒック出現量を演算で求め、単位時間に発生しているトラヒックの量を推定する。図4の例では IP ルータを経由するトラヒックの内訳として APL1 から APL4 が存在し、通信フローの観測によりそれぞれの流量が可視化される。

大規模通信網における観測データ量を削減するためには、サンプリング周期をある程度長く設定する必要があるが、この場合観測精度が低下するため、通信フローのカウントに代わる新たな観測方法が研究されている。これらの新たなフロー観測法の基本的な発想は、確率的データ構造を用いて通信フローの情報量を削減する点にある。例えば、Liu らは確率的データ構造である Count-min-sketch 構造を用いた通信フロー観測法を提

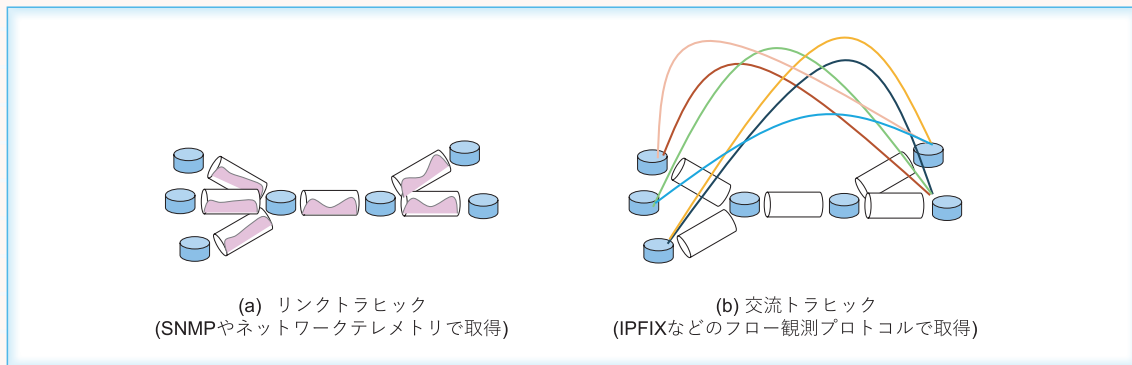


図3 リンクトラヒックと交流トラヒックの違い

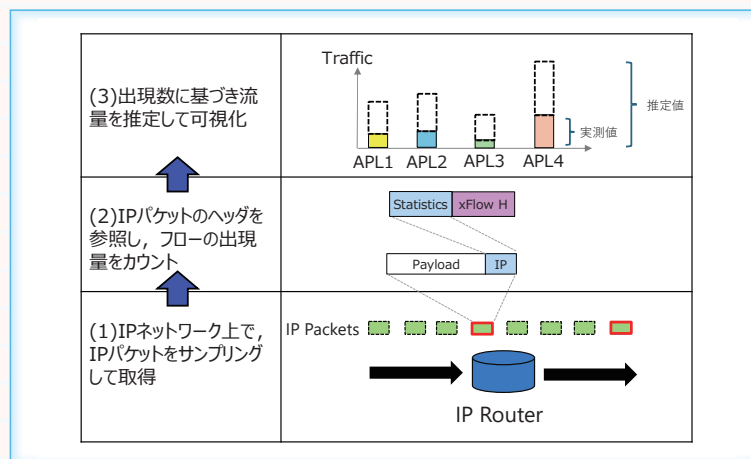


図4 通信フロー観測プロトコルの動作例

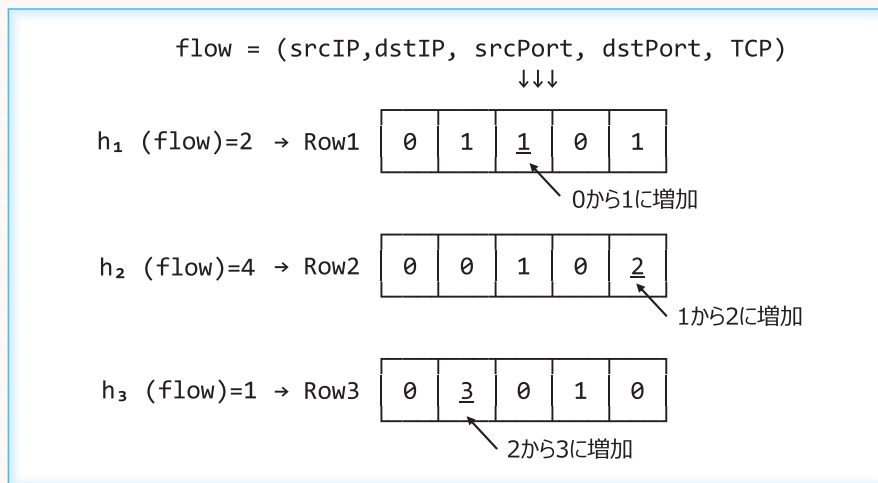


図5 Count-min-sketch の動作例

案している⁽⁶⁾。Count-min-sketch は高次元データをハッシュにより、より小さな空間である確率的データ構造へと射影するデータ構造である。

図5に例を示すように、複数のハッシュ関数 (h_1 , h_2 , h_3) とそれぞれに対応する配列 (Row1, Row2, Row3) から構成される。まずハッシュ関数 h_1 , h_2 , h_3 を用いて、通信フロー情報から異なる三つのハッシュ値 ($h_1=2$, $h_2=4$, $h_3=1$) を計算する。次に、このハッシュ値を添え字として、Row1[2], Row2[4], Row3[1] の値をそれぞれ1増加させることで、フローの出現量を各配列上に記憶する。ただしこの例では、Row2[4] と Row3[1] はハッシュ値の衝突、すなわち既存のフローの出現がカウントされている想定であり、それぞれ Row2[4]=2, Row3[1]=3 となる。Count-min-sketch ではこの衝突の影響を最小限とするために、各配列の値から最小値を取得する。図5の例では、最小値であり衝突の発生していない Row1[2]=1 が該当フローの出現量として採用される。こうして、通信フロー情報を格納するために、本来必要となるデータサイズを削減しつつ、衝突の影響を抑え精度高く情報を保持する工夫を行っている。

3

時系列トラフィックデータを用いた異常検知

本章ではトラフィック活用の事例として、ネットワークにおける異常検知を目的とした時系列データ分析手法の分類について述べる。次に過去のデータとの相関分析に基づく異常検知手法について解説する。

3.1 時系列データ分析手法の分類

時系列データを用いた異常検知手法として、時系列データをモデル化し、モデルから予測された値と実際に

観測される値とを比較して異常を検知していく方法が提案されている。例えば、過去のデータの自己相関やトレンドをモデル化し、その関係式から統計的な予測を行う ARIMA (Autoregressive Integrated Moving Average)⁽⁷⁾ や、時系列データの依存や非線形なパターンをニューラルネットワークによって学習し、複雑なデータの動きを柔軟に予測する LSTM (Long Short-Term Memory)⁽⁸⁾、また局所的な時系列パターンを畳込みで捉えて値を予測し、その予測誤差から異常を検出する CNN (Convolutional Neural Network)⁽⁹⁾ を用いた方法などがある。

一方、将来の値を予測するのではなく、過去の傾向から正常なトラフィックパターンの特徴量を抽出し、観測された値を含む特徴量と比較して、大きく差異があれば異常と判定する方法も提案されている。データの扱いを容易にするために、特徴量は元のデータと比較して次元削減などの工夫が行われ、例えば主成分分析 (PCA) や、スペクトル変換などの方法が利用される⁽¹⁰⁾。

一般的に、多くのユーザ通信が重畳される大規模キャリアネットワークにおいては、故障などの外部要因が作用しない限りは、トラフィックの時系列パターンは単調増加しつつも周期的な変動傾向を有すると考えられる。そこで我々は、将来の値を予測することなく、シンプルな相関値解析によって高速かつ精度良く異常検知を行う方法を提案してきた⁽²⁾。次節ではこの方法を紹介する。

3.2 相関分析による異常検知の例⁽²⁾

大規模な広域 IP ネットワークにおいては、トラフィックの時系列パターンは周期的な変動傾向を有しており、その傾向が変化した場合に異常が発生したものと想定される。この想定の下で、異常検知をシンプルに実現する方法を紹介する。

紹介する異常検知手順では、分析対象の通信フローの

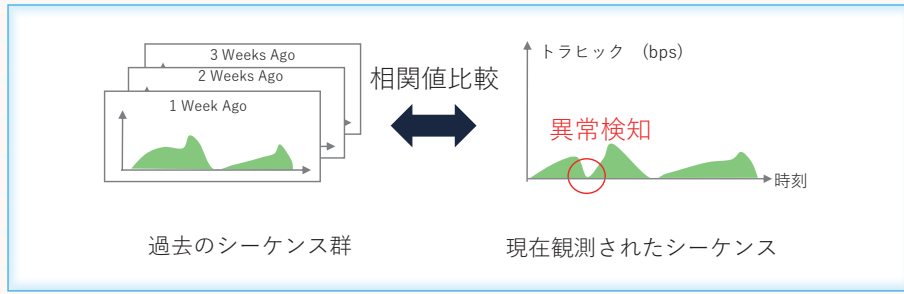


図6 相関値を用いた異常検知の概要

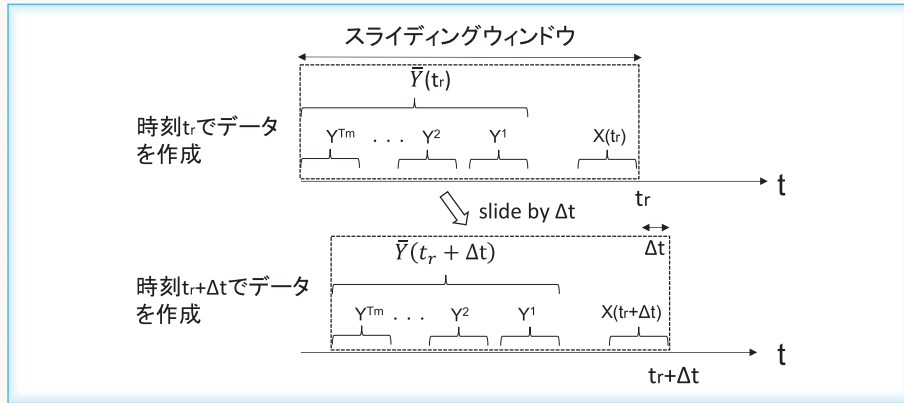


図7 スライディングウィンドウ

観測時点における形と、その通信フローの過去の形との相関値を計算することで異常発生の有無を判断する（図6）。ここで時系列データがある範囲で区切ったものを「シーケンス」と呼び、分析対象とするシーケンス X を以下のとおり作成する。

$$X = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n \quad (1)$$

各要素は観測する通信フローの流量で、単位は bit/s である。例えば、通信フローの1日の変動を分析する場合、1時間に1回通信フローを観測できるものとする、シーケンスのサイズ n は24となる。次に、分析対象の T 週間前のシーケンス Y を以下のとおり作成する。

$$Y = (y_1^T, y_n^T, \dots, y_n^T) \in \mathbb{R}^n, \quad T = 1, 2, \dots, T_m \quad (2)$$

ここで、過去の観測データが特異点である影響を抑えるために、 T_m 個の過去の観測データを用いて、平均値 $\bar{Y}$ を計算する。

$$\bar{Y} = (\bar{y}_1, \bar{y}_2, \dots, \bar{y}_n) \in \mathbb{R}^n \quad (3)$$

ここで $\bar{y}_l$ は以下のとおりである。

$$\bar{y}_l = \frac{1}{T_m} \sum_{k=1}^{T_m} y_l^k \quad (4)$$

シーケンス X とシーケンス $\bar{Y}$ の次元は同じであるから次式により相関値 $\rho_{X, \bar{Y}}$ を計算可能である。

$$\rho_{X, \bar{Y}} = \frac{\frac{1}{n} \sum_{t=1}^n (x_t - \bar{x})(y_t - \bar{y})}{\sigma_x \cdot \sigma_y} \quad (5)$$

ここで、 $\bar{x}$ 、 $\bar{y}$ はそれぞれ X 、 $\bar{Y}$ の平均で σ_x 、 σ_y はそ

れぞれ X 、 $\bar{Y}$ の標準偏差である。 $\rho_{X, \bar{Y}}$ は標準化された値であるから、二つのシーケンスが連動しているほど +1 に近い値をとり、連動がない場合は 0 に近い値となる。 $\rho_{X, \bar{Y}}$ があらかじめ決定したしきい値以下となった場合は、シーケンス X において何かしらの異常が発生したものと判断される。

この異常検知手順はシーケンス X 及び $\bar{Y}$ が与えられた上で計算を行う方法であるため、準リアルタイムに異常検知を行うためにスライディングウィンドウを利用する。スライディングウィンドウは図7に示すように、最新のトラヒック観測時刻 t_r を終点としたシーケンス $X(t_r)$ 及び $\bar{Y}(t_r)$ を上記の手順で作成し、相関値 $\rho_{X(t_r), \bar{Y}(t_r)}$ を計算する。次に観測時間 Δt 経過後の時刻 $t_r + \Delta t$ におけるシーケンス $X(t_r + \Delta t)$ 及び $\bar{Y}(t_r + \Delta t)$ を作成し、同様に相関値 $\rho_{X(t_r + \Delta t), \bar{Y}(t_r + \Delta t)}$ を計算する。このように Δt の時間間隔で分析データを用意し相関値を計算することで、 Δt の間隔で異常検知の判定が可能となる。 Δt を小さくすることで、より短い間隔での異常検知が可能となるが、その分、データの記録や計算のオーバーヘッドが増加する。

文献(2)では、急激なトラヒック増加やサービス故障などのユースケースにおいて、通信フローの流量の変化に対する異常検知の精度が議論されている。通信フローの流量の変化の大きさはユースケースによって異なるため、常に固定のしきい値で正確な異常検知を行うことは難しい。そこで、しきい値を外部条件に応じて動的に変

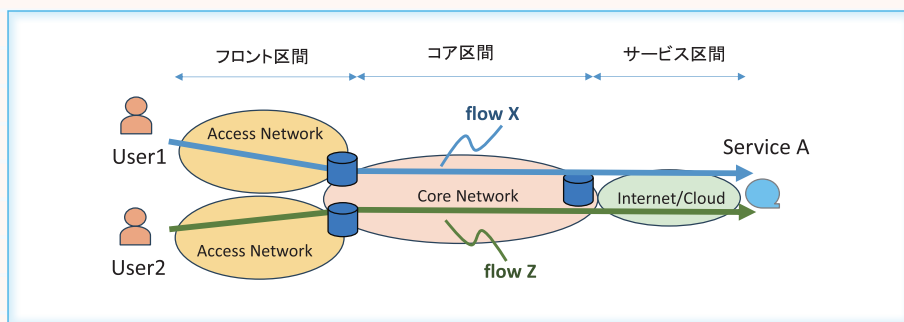


図8 対象とするネットワーク

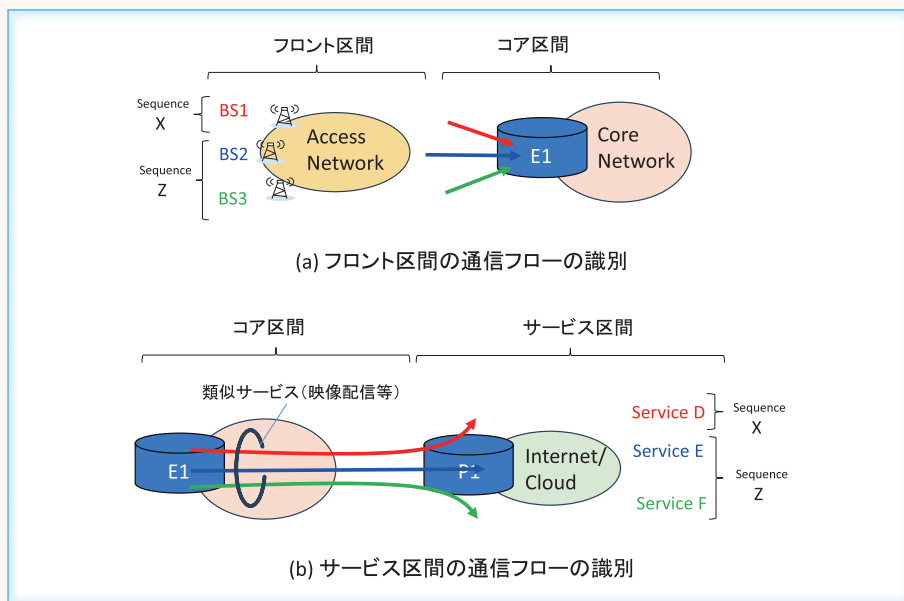


図9 ネットワークと通信フローのモデル

化させる方法を提案し、その有効性が報告されている。

4 異常箇所推定への応用

3. で解説した異常検知方法は、分析対象の通信フローに異常が起きたかどうかのみを判定しており、その原因となる障害箇所は特定できない。一般的に大規模なキャリアネットワークを介したインターネットアクセスは、ユーザと基地局や通信局舎を結ぶアクセスネットワーク、日本全域を接続するコアネットワーク、及びインターネット事業者やクラウド事業者が提供するネットワークから構成される(図8)。素早い故障対応オペレーションを行うためには、上記のいずれのポイントで障害が発生しているかを検知できることが重要である。

本章では通信フローを観測して分析することで、異常の有無を検知するとともに、上記のネットワークを (i) フロント区間, (ii) コア区間, 及び (iii) サービス区間と定義したときに、どの区間において異常が発生しているかを推定する方法について解説する。

4.1 ネットワークモデルの定義

図9を用いてネットワークモデルを説明する。図9(a)ではフロント区間とコア区間が定義されており、基地局BS1からコアルータE1へ流入する通信フローが分析対象のシーケンスXとなる。このとき、空間分析を行うためにBS1以外(図ではBS2とBS3)からの通信フローも観測し、それぞれシーケンスZとして次式で定義する。

$$Z^S = (z_1^S, z_2^S, \dots, z_n^S) \in \mathbb{R}^n, S = 1, 2, \dots, S_m. \quad (6)$$

シーケンスZは S_m 個の基地局から発生しているフローの数だけ観測されるため、次式のとおりに平均値 $\bar{Z}$ を求める。

$$\bar{Z} = (\bar{z}_1, \bar{z}_2, \dots, \bar{z}_n) \in \mathbb{R}^n, \quad (7)$$

ここで $\bar{z}_l$ は次式で与えられる。

$$\bar{z}_l = \frac{1}{S_m} \sum_{k=1}^{S_m} z_l^k. \quad (8)$$

コア区間とサービス区間においても同様に、分析対象のシーケンスXとは別に、異なるサービスを利用する通信フロー群をシーケンスZとして観測する。ただし、

```

01: For incoming stream X
02:   Create  $\bar{Y} = (\bar{y}_1, \bar{y}_2, \dots, \bar{y}_n) \in \mathbb{R}^n$ 
03:   Create  $\bar{Z} = (\bar{z}_1, \bar{z}_2, \dots, \bar{z}_n) \in \mathbb{R}^n$ 
04:   IF  $s1 := \text{Similarity}(X, \bar{Y}) < Th1$ 
05:     X-Y_Error := true
06:   IF  $s2 := \text{Similarity}(X, \bar{Z}) < Th2$ 
07:     X-Z_Error := true
08:   SetAlert(X-Y_Error, X-Z_Error, s1, s2)
09: Exit()

```

図 10 異常箇所推定のアルゴリズム

X と Z は、例えばいずれも映像配信サービスであるといった、同様のトラフィック変動傾向を有する類似のサービスであることを想定している。

異なる複数の基地局から特定のコアルータに流入する通信フローや、特定の送信元から異なるサービスへ向かう通信フローは、全て通過するコアルータ上で観測可能である。なお、一般的なキャリアネットワーク上では、発生している通信フローの数が膨大であり、更に諸般の事情により特殊なカプセル化 (TCP/IP ヘッダ情報の隠蔽) が行われることがあるが、筆者らが開発した装置⁽¹¹⁾により、通信フローは詳細に観測可能であるものとする。

4.2 時空間分析による異常箇所推定アルゴリズム

図 10 に異常箇所を推定するアルゴリズムの擬似コードを示す。アルゴリズムの目的は、分析対象の通信フローの自己相関に差異があるかどうかを示す $X-Y_Error$ 及び、分析対象の通信フローと類似のサービスによる通信フロー群との間に差異があるかどうかを示す $X-Z_Error$ の状態を決定することである。図 10 では分析対象のシーケンス X に対し、まずシーケンス Y と Z を作成する (line 2-3)。次に X と Y の類似度を取得し、それがしきい値以下であれば $X-Y_Error$ を真とする (lines 4-5)。類似度の計算を行う $\text{Similarity}()$ 関数は 3. の式 (5) を利用可能である。同様に X と Z の類似度を取得し、しきい値未満であれば $X-Z_Error$ を真とする (lines 6-7)。最後に $X-Y_Error$ 及び $X-Z_Error$ の値を用いて $\text{SetAlert}()$ 関数によって障害の有無、及び障害位置を判定する (line 8)。

続いて $\text{SetAlert}()$ 関数の仕様を図 11 を用いて説明する。 $\text{SetAlert}()$ 関数には $X-Y_Error$ 及び $X-Z_Error$ の値が入力される。どちらも False の場

Case	入力		出力		
	X-Y Error	X-Z Error	コア区間異常	X の異常	X 以外の異常
(i)					
(ii)	X		被疑箇所		
(iii)	X	X		被疑箇所	
(iv)		X			被疑箇所

図 11 $\text{SetAlert}()$ 関数の動作

合 (case (ii)) は、 X 自身の振る舞いは過去と相違がなく、また、ほかの通信フローと同様の振る舞いを行っていることから、異常区間は「なし」と判定される。 $X-Y_Error$ のみがある場合 (case(ii)) は、 X 自身の振る舞いは過去と相違があるものの、同時刻のほかの通信フローとは相関がある状態を意味している。つまり、分析対象時刻においては、ほかの通信フローも含めて異常が発生していると類推されるため、コア区間において障害が発生していると判断する。次に $X-Y_Error$ と $X-Z_Error$ の両方が発生している場合を考える (case(iii))。このケースでは X 自身の振る舞いは過去と相違があり、かつ同時刻の通信フローとも相関が低いことから、 X 自身異常が発生していると判断する。最後に $X-Z_Error$ のみが発生している場合を考える (case(iv))。このケースでは、 X 自身の振る舞いは過去と相違がないが、ほかの通信フローとの相関が低いことから、ほかの通信フロー群のいずれかの通信フローにおいて異常が発生していると判断される。この場合は、ほかの通信フロー群に同様のアルゴリズムを順次適用することで、被疑フローを特定可能である⁽¹¹⁾。なお、 $\text{SetAlert}()$ 関数は式 (5) で得られる相関値 $s1, s2$ を受け取ることで、被疑箇所の推定確率を示すように拡張も可能である。

この方法を用いて、監視するフローを切り換えることでネットワーク全体の故障箇所判定が可能となる。例えば、図 9(a) のようにフロント区間のフローを監視する場合は、異常区間がコア区間であるのか、基地局 (若しくは有線網の局舎ビル) であるのかが判定される。図 9(b) のようにサービス区間のフローを監視する場合は、異常区間がコア区間であるのか、サービス区間であるのかが判定される。これらのフロー情報はいずれもコア区間に設置されるコアルータ (図 9 の E1) で観測可能である。よって、コア区間を運用する通信事業者は自身の

管理下にあるルータ上の通信フローを観測するだけで、フロント区間、コア区間、サービス区間のどこで異常が発生しているのかを一元的に判定することが可能となる。

5 おわりに

本稿ではトラフィック情報の観測方法を述べた後に、その活用事例として、トラフィックの時間的及び空間的な変化を解析することで、情報通信ネットワークにおける異常発生の有無や、異常発生を位置を推定する方法について解説してきた。これまで述べたとおり、トラフィックは安定した情報通信ネットワークの運用のために重要な情報である一方で、その観測自体が困難な場合もある。例えば、インターネットトラフィックは年々指数関数的に増加している一方で、その観測容量は様々な制約により十分な容量を確保できないケースも存在する。このため、確率的データ構造を活用した方法⁽⁶⁾や、収集する位置や種別の最適化⁽¹²⁾など、今後もトラフィック観測手法の改善を行っていくことが重要である。

文献

- (1) 植松芳彦, 鎌村星平, 杉山隆太, 武田知典, 宮村崇, 笹山浩二, “交流トラフィック推定技術の大規模サービス統合IP網への適用性,” 信学論(B), vol. J98-B, no. 3, pp. 255-265, March 2015.
- (2) S. Kamamura, Y. Takai, M. Nishimura, Y. Hayashi, and T. Fujiwara, “Network anomaly detection through IP traffic analysis with variable granularity,” IEEE Access, vol. 11, pp. 129818-129828, Nov. 2023.
- (3) 吉田友哉, 鈴木崇史, 松崎光成, “大会期間中のイベントがインターネットトラフィックに及ぼした影響の一考察——トラフィックとテレビ視聴率相反関係について,” 信学誌, vol. 105, no. 8, pp. 956-960, Aug. 2022.
- (4) M. Yu, “Network telemetry: towards a top-down approach,” ACM SIGCOMM Comput. Commun., vol. 49, no. 1, pp. 11-17, Feb. 2019.
- (5) B. Claise, B. Trammel, and P. Aitken, “Specification of the IP flow information export (IPFIX) protocol for the exchange of flow information,” IETF RFC7011, Sept. 2013.
- (6) Z. Liu, A. Manousis, G. Versaeger, V. Sekar, and V. Braverman, “One sketch to rule them all: rethinking network flow monitoring with univMon,” Proc. SIGCOMM, pp. 101-114, Aug. 2016.
- (7) F. Wang, M. Li, Y. Mei, and W. Li, “Time series data mining: A case study with big data analytics approach,” IEEE Access, vol. 8, pp. 14322-14328, Jan. 2020.
- (8) P. Malhotra, L. Vig, G. Shroff, and P. Agarwal, “Long short term memory networks for anomaly detection in time series,” Proc. ESANN April 2015.
- (9) M. Munir, S. A. Siddiqui, A. Dengel, and S. Ahmed, “Deep-AnT: A deep learning approach for unsupervised anomaly detection in time series,” IEEE Access, vol. 7, pp. 1991-2005, Dec. 2018.
- (10) H. Huang et al., “Network traffic anomaly detection based on principal component analysis and clustering,” Proc. INCoS 2014.
- (11) S. Kamamura, Y. Hayashi, Y. Miyoshi, T. Nishioka, C. Morioka, and H. Ohnishi, “Fast xFlow proxy: exploring and visualizing deep inside of carrier traffic,” IEICE Trans. Commun., vol. E105.B, no. 5, pp. 512-521, May 2022.
- (12) 鎌村星平, 平田孝志, “通信網の状態に応じた適応型トラフィック監視法に関する検討,” 信学技報, NS2024-124, pp.13-18, Nov. 2024.

鎌村星平 (正員: シニア会員)

平 16 早大・理工・電子・情報通信卒。
平 18 同大学院国際情報通信研究科修士課程了。平 25 同大学院博士課程了。
平 18 日本電信電話株式会社入社。令 3 成蹊大准教授。令 7 同大学教授。
現在、情報通信ネットワークの研究に従事。博士 (国際情報通信学)。



通信障害対処をクラスタリング分析によって自動化するネットワーク運用保守管理支援技術

Network Operation and Maintenance Support Technology for Automating Countermeasures against Communication Failures Using Clustering Analysis

三宅裕士 Yuji Miyake[†] 角 武憲 Takenori Sumi[†] 末廣 雄 Takeshi Suehiro[†]

Summary

無線ネットワークシステムの複雑化に伴い、ネットワークの運用・保守管理の自動化が求められている。特に、通信障害対処においては、従来、複数の機器にまたがる運用データを収集し、無線通信・ネットワークに関する有識者が分析を行い、障害要因の特定及び対策の検討を行っていた。通信障害対処を自動化するためには、運用データの収集、障害要因分析、対処案の検討及び実行のそれぞれの機能の自動化を図る必要がある。そこで、本稿では、通信障害対処に関する運用データの収集・分析・対処の自動化に資する技術及び実証システム試作について解説する。

Key Words

通信障害、クラスタリング分析、ネットワークテレメトリー、ネットワーク運用・保守管理

1 はじめに

無線ネットワークシステムを用いてサービスを提供するアプリケーションが多様化し、また、無線ネットワークシステムに適用される技術が高度化するにつれ、構成する機能要素や装置が多くなり、無線ネットワークシステムは複雑化する傾向にある。そのため、近年ネットワーク分野における運用や保守に関する自動化・最適化技術に関する研究開発が行われている^{(1)~(3)}。

特に、無線ネットワークシステムのサービス継続性を担保するためには、無線ネットワークの通信障害時の障害検知から対処実行までの過程への対応が重要である。無線ネットワークシステムで通信障害が発生した際は、複数の装置からログなどの運用データを収集し、無線通信はもちろん、ネットワークに関する知見を有する熟練者が分析を行い、障害要因について切り分け、更に障害対策を検討する、というように、障害対処は時間を有するフローであると言える。今後、様々なユースケースにおいて無線ネットワークシステムが活用されること、無線ネットワークシステムが単一アプリケーションで使用される専用システムではなく、複数のアプリケーションを柔軟に収容すること、更に通信・ネットワークに関する熟練者不足への対策、あるいは運用・保守

管理の省力・省人化を実現するために、通信障害対処に関するフローを自動化または半自動化する必要があると考える。

本稿では、ネットワークの運用・保守管理の自動化技術の中でも、通信障害時の障害検知から対処実行までのフローを半自動化するために必要となる技術要素について説明し、実験室レベルで通信障害から復旧までの実証を行った結果を示す。

本稿の構成は以下のとおりである。**2.**では通信障害発生時に対策を自動化するための技術要素を説明する。**3.**では通信ネットワーク及びその構成機器の状況をモニタリングするネットワークテレメトリーについて説明する。**4.**では通信障害の要因分析を行うクラスタリング分析について説明する。**5.**では、実験室レベルでの実証システム構築及び実証結果を説明する。最後に**6.**でまとめを述べる。

2 ネットワーク運用・保守管理の自動化のための技術要素

ネットワークに関する自動化・自律化については、3GPP (3rd Generation Partnership Project) や ETSI (European Telecommunications Standards Institute) の標準化グループ (ISG: Industry Specification Group) である ENI (Experimental Networked Intelligence) など様々な標準化団体において議論されている。表1は、通信分野全体にわたる国際標準化団体である TM Forum にて定義されている自動化の実現レベルと、

[†] 三菱電機、鎌倉市
Mitsubishi Electric Corp., Kamakura-shi, 247-0056 Japan

表1 TM Forum における Autonomous Networks Levels

レベル 定義	L0: Manual Operation & Maintenance	L1: Assisted Operation & Maintenance	L2: Partial Autonomous Network	L3: Conditional Autonomous Network	L4: High Autonomous Network	L5: Full Autonomous Network
実行	P	P/S	S	S	S	S
認知	P	P	P/S	P/S	S	S
分析	P	P	P	P/S	S	S
決定	P	P	P	P	P/S	S

ネットワーク運用において必要となる要素との関係を示したものである⁽⁴⁾。表中の P (Personal) は人手によって、S (System) は自動化システムによって対処が行われることを表しており、これまで人手で行ってきた「実行・認知・分析・決定」のプロセスの段階的な自動化が想定されている。よって、そのための仕組み・技術が求められている。

表1中の認知、分析及び決定に着目すると、通信障害対策の場合のフローは、検知、障害箇所特定、要因分析、対策立案、判断となる。本稿では、これらのうち、判断以外を自動化することで、運用者の効率的な障害対処を実現することを想定する。以下に本稿で解説する通信障害対処を実現する技術要素を示す。

① ネットワークテレメトリー

ネットワークの状態やパフォーマンスの監視・分析のために生成・収集・蓄積・分析されるデータ、あるいはそのプロセスである。

② 障害要因分析

テレメトリーなどによって無線ネットワークシステムの各機能・各装置から収集した運用データを用い、自律的に通信障害・品質劣化の要因分析を行う。本稿では、クラスタリング分析を用いた障害要因分析手法について紹介する。

③ ネットワーク運用・保守管理基盤

無線ネットワークシステムの運用者に対して、稼働状況や運用状況を提示する、また、障害要因分析結果の提示及び障害要因分析結果に基づく対策案を提示し、無線ネットワークシステムの運用を支援する。

3 ネットワークテレメトリー

ネットワークテレメトリーとは、ネットワークの状態やパフォーマンスの監視・分析を目的として、ネットワーク機器において生成し、収集・蓄積・分析されるデータ、あるいはそのプロセスである。従来、特定のアプリケーション向けに設計・構築され

た準静的なネットワークを人間が手動で運用することを前提として、SNMP (Simple Network Management Protocol) や Syslog に代表される、必要時にネットワーク機器にアクセスして機器やネットワークの情報を取得したり、人間の運用者が読むためのログ／イベント情報を収集したりする方式が用いられてきた。しかし、近年、5G (5th Generation Mobile Communication System) をはじめとして、多様な要件のサービスを収容可能なネットワークシステムが登場し、また、SDN (Software Defined Networking, ソフトウェア定義ネットワーク) の導入が進み、柔軟かつ動的なネットワーク制御が可能となってきたことから、その監視・分析及び制御自動化／効率化／最適化のために、より詳細かつリアルタイムなテレメトリーデータの取得と解析が求められるようになった。

その結果、解析は機械学習によるビッグデータ処理での実現が期待される一方で、取得についても新たなテレメトリー技術が提案されている。従来のテレメトリー技術と異なる特徴として、大量のテレメトリーデータを高速に取得するためにプッシュ型若しくはストリーミング型で送信されること、多様な機器からのデータを統合的に分析できるようにモデル化・正規化されていること、ネットワーク運用の一部として効率化・最適化するために取得の頻度や対象とする情報を動的かつ対話的に制御できることなどが挙げられる。ここでは、通信ネットワークの経路上でアプリケーションのデータパケットにメタデータを付加することで、パケット単位でリアルタイムにネットワークや機器の状態に関する情報を取得する INT (In-band Network Telemetry)⁽⁵⁾を紹介する。

図1にINTの動作概要を示す。アプリケーションのデータパケットを送信する sender から受信する receiver までのネットワーク経路上の INT に対応する各ノードにおいて、テレメトリーの対象とするアプリケーションの通信フローの各パケットに対し、通常のパケット転送処理と並行して、タイムス

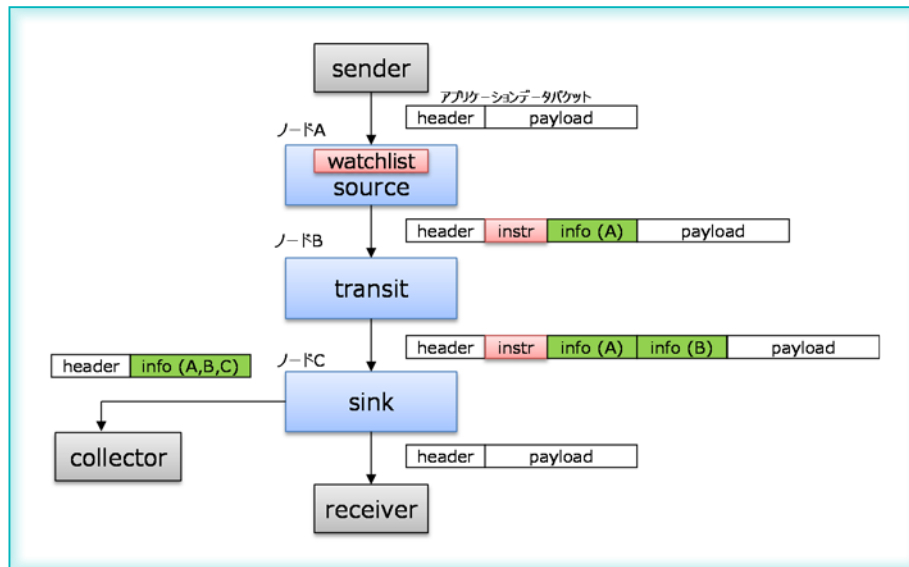


図1 ネットワークテレメトリの概要

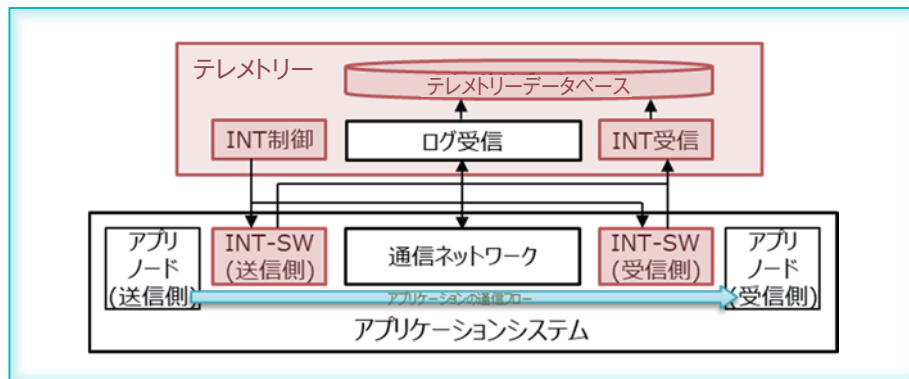


図2 テレメトリシステムの構成例

タンプ、転送処理に要した時間、バッファ使用量、リンク使用率などの情報を取得し、メタデータ（図1中のinfo(A)など）としてパケットに付加していき、「sink」と呼ばれるネットワークの出口に位置するノードで、アプリケーションパケットから取り外したメタデータをテレメトリコレクタ（図1中のcollector）に送信する。どのフローに対して何のデータを取得するかは、「source」と呼ばれるネットワークの入口に位置するノードでsource instruction（図1中のinstr）としてパケット内に埋め込むか、または、各ノードにwatchlistとして設定する方法がある。

現在、広く一般に用いられているネットワーク機器は上述のような処理を行うINT機能をサポートしていないが、スクラッチで設計・実装することが必須ということはない。ネットワーク機能のデータプレーン処理を記述するプログラミング言語として策定されているP4⁽⁶⁾は、INTを代表的なアプリケーションの一つとして想定しており⁽⁷⁾、P4対応デバイスであれば、P4言語で記述したINT機能を各

デバイス向けにコンパイルしてロードすることでINTを実現可能である。（ただしP4対応デバイスが全てのP4規定をサポートするわけではない点には注意が必要。）P4は、当初は限定されたスイッチ装置のみが対応していたが、スマートNICやソフトウェアスイッチに広がっており、今後、更なる普及が期待される。

図2に通信ネットワーク全体にわたるテレメトリを取得するテレメトリシステムの例を示す。通信ネットワークの両端にINT機能を持つスイッチが配置され、アプリケーションの通信フローごとに、例えば、シーケンス番号、タイムスタンプ、累積バイト数をINTデータとして埋め込むようにすれば、アプリケーションレベルで通信ネットワークにおいて発生するパケットロスや、各パケットの到達時間の揺らぎによる並び替えの検出、遅延、スループットをリアルタイムに計測可能である。

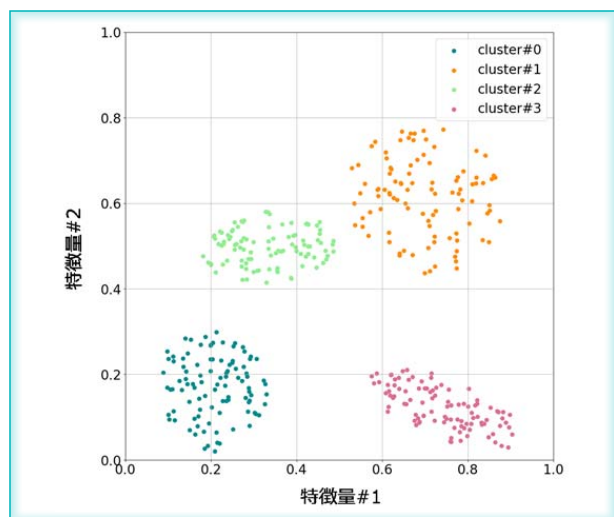


図3 k-means によるクラスタリング例

4 クラスタリングによる障害要因分析

本章では、無線ネットワークシステムで発生した障害要因を推定するための分析技術について説明する。

機械学習の適用により障害検知を行う技術としては、サポートベクターマシンや、クラスタリング、ニューラルネットワークなどの手法が用いられる^{(8)~(11)}。様々な無線ネットワークシステムや環境に適用することを想定すると、サポートベクターマシンのような「教師あり学習」を用いた機械学習手法の場合、ほかの無線ネットワークシステムへの適用時に学習データへのラベル付け作業（アノテーション）が必要であり、学習データの事前処理にコストが掛かる。そのため、本稿では「教師なし学習」が可能なクラスタリング手法を用いた障害要因分析に着目し、教師なしでデータのクラスタリングを行う k-means を用いた事例を解説する。

k-means は、データを複数のクラスタに分類するための代表的な手法である⁽¹²⁾。この手法は、各クラスタの中心点を計算によって決定し、各データに対して、決定したクラスタの中心点までの距離を算出し、最小距離を示すクラスタに分類する。k-means は、そのシンプルさと計算効率の高さから、広く利用されている機械学習手法である。以下にその手順を示す。

① 初期化

複数のデータの中からランダムに k 個のデータを選択し、クラスタの中心点とする。この初期化ステップは、アルゴリズムの結果に大きな影響を与えるため、適切な方法で行うことが重要である。

② 割当てステップ

表2 障害要因分析に用いた運用データ

項目	取得方法
所望信号受信電力	無線装置ログ
干渉電力	無線装置ログ
遅延時間	テレメトリ
送信スループット	テレメトリ
受信スループット	テレメトリ

表3 クラスタ分類

Cluster	状態	判定
#0	正常状態	正常状態
#1	ほかのトラヒック入力により受信スループット低下は大きくないが、遅延時間が増加	アプリレイヤでのトラヒックふくそう（影響小）
#2	ほかのトラヒック入力による受信スループットの低下と、遅延時間の増加	アプリレイヤでのトラヒックふくそう（影響大）
#3	干渉信号による、フレーム誤りの発生	無線区間での干渉

次に、各データを最も近い中心点のクラスタに割り当てる。このプロセスにより、データは k 個のクラスタに分類される。

③ 更新ステップ

各クラスタに割り当てられたデータの平均位置を計算し、各クラスタの中心点を更新する。この更新ステップにより、中心点はクラスタの適切な中心点に近づく。

④ 手順②、③を繰り返す

割当てステップと更新ステップを繰り返し実行する。中心点の位置が変わらなくなるか、事前に設定された最大回数に達するまでこの処理を続ける。

k-means を用いることで、図3に示すように、教師なしで特徴量に基づいて、各データがどのクラスタに属するかを判定することができる。図3は二つの特徴量を考慮したクラスタリングの例である。これらのクラスタがどういった意味を持つかは、各クラスタにおける特徴量を基に決定することができる。クラスタリングによる障害要因分析では、各クラスタを作成するときの特徴量を通信ログから決定し、それぞれがどういった意味を持つクラスタかをクラスタリング結果を確認し決定する。

本稿では、アンライセンズバンドを用いる無線ネットワークシステムを想定し、表2に示す無線装置における無線通信に関する運用データ及びテレメトリにより取得したトラヒック情報から特徴量を

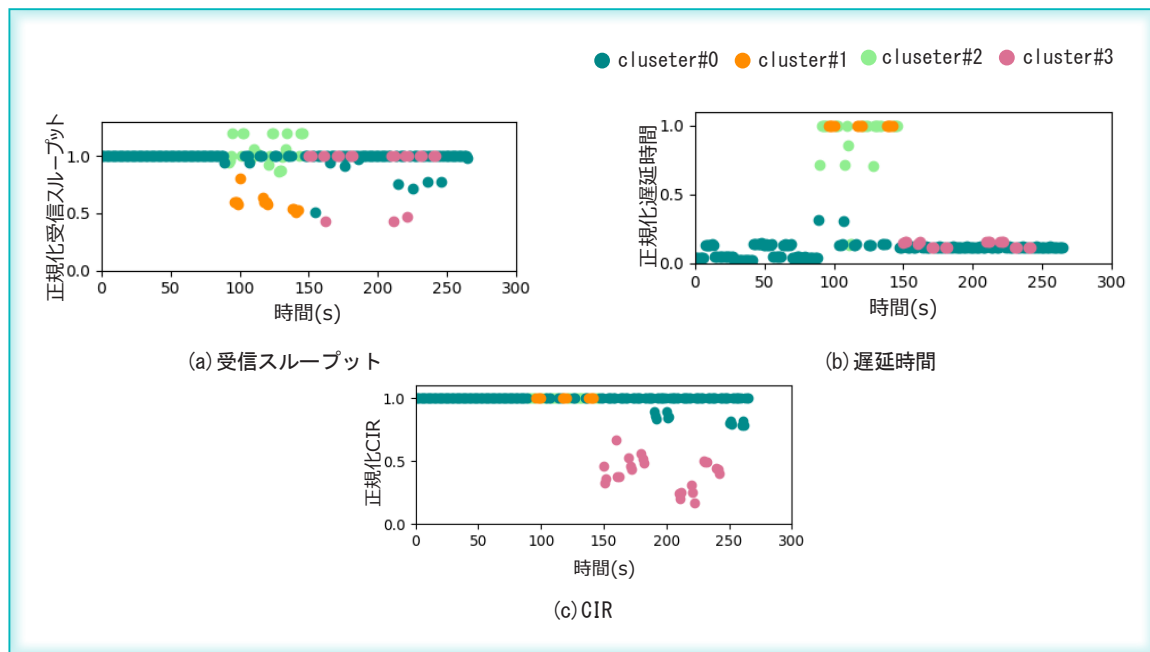


図4 無線ネットワークシステムの運用データのクラスタリング分析結果

決定し、k-means によるクラスタリングを試行した結果を示す。また、k-means では、各データのユークリッド距離を基にクラスを決定した。そのため、特定の特徴量のみ、大きな値を取るようなことがあれば、正常にクラスが形成されなくなってしまう。これを避けるために、各データの値は正規化しておく必要がある。例えば、干渉の場合は、所望信号受信電力から CIR (Carrier to Interference Ratio) を算出し、規定の BER (Bit Error Rate, ビット誤り率) を得ることができる CIR を 1 と設定し、規定の BER を超え、通信できない状態となる CIR を 0 となるように各データの値を調整する。表 3 に分類したクラスの内容を示す。

無線ネットワークシステムの障害要因として、無線区間の受信信号電力の低下によるスループットの低下やアプリケーションでのトラヒックふくそうに加え、アンライセンズバンドを用いる無線ネットワークシステムの場合、同一周波数を用いるほかの無線ネットワークシステムからの干渉を受けることによる通信障害も考えられる。図 4 に無線ネットワークにおける無線装置及びテレメトリーで収集した運用データのうち、正規化した受信スループット及び遅延時間、CIR の時系列データを示す。受信スループットは、送信スループットに対する受信スループットの割合を示している。遅延時間は、対象とする無線ネットワークシステムの許容時間を 1 となるように正規化した。また、CIR は、無線ネットワークシステムの通信性能を満足する CIR が 1 となるように正規化している。図中の各色のマッピングは、

障害要因分析を行った結果である。

各グラフの横軸は時間を示す。運用データの変化点・特徴量に対してクラスタリング分類が動作していることが確認できる。例えば、ネットワークテレメトリーにより計測された遅延時間が大きくなった場合（横軸：100～150 秒）において、CIR が低下していないことから無線区間の干渉 (cluster #3) に分類されず、アプリケーションでのトラヒックふくそう (cluster #1, #2) に分類されていることが確認できる。また、受信スループットの低下度合いに応じ、トラヒックふくそうの影響度が異なる二つのクラスに分類されていることが確認できる。このように多角的な運用データを用いたクラスタリング分析により、無線ネットワークシステムにおける障害要因分析を行うことが可能となる。

5

通信障害の対処に関する実証システム試作

本章では、これまで説明した運用データを収集するテレメトリーシステム及び障害要因分析を実現するクラスタリング分析に加え、無線ネットワークシステムの運用状況の可視化及び障害対策提案を行うネットワーク運用・保守管理基盤を試作し、実証した結果について説明する。試作したネットワーク運用・保守管理基盤は「NW (Network) エンジニアリングツール」と呼ぶ。図 5 に試作した実証システムの構成を示す。実証システムでは、無線ネットワークシステムを用いた監視カメラシステムを模擬

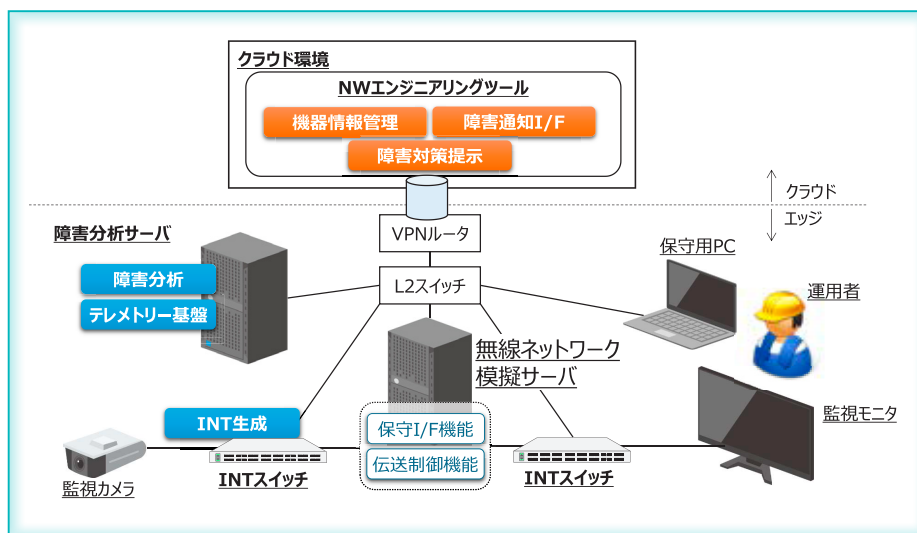


図5 試作した実証システム

NWエンジニアリングツール			
■ 装置状態情報			
装置ID	装置種別	属性情報	説明
1000	高信頼無線	高信頼無線装置（親）	基地局(生産エリアA)
1001	高信頼無線	高信頼無線装置（子）	固定カメラ(植立ラインA監視)接続無線装置
2000	高信頼無線	高信頼無線装置（親）	基地局(生産エリアB)
2001	高信頼無線	高信頼無線装置（子）	移動カメラ(生産状況監視)接続無線装置
2002	高信頼無線	高信頼無線装置（子）	貨物運搬AGV#1接続無線装置
2003	高信頼無線	高信頼無線装置（子）	植立AGV#1接続無線装置
追加 削除			
■ 障害通知			
日時	装置ID	属性種別	障害内容
2025-02-20 17:04:22	1001	高信頼無線装置（子）	アプリレイヤでのトラヒック経路が発生しました

図6 NW エンジニアリングツール GUI

した。無線ネットワークシステムを介して、監視カメラ映像トラヒックを伝送し、運用者に監視カメラ映像を提供する。本試作では、無線ネットワークは、汎用サーバ内の仮想環境にて送受信装置の伝送制御動作及び各保守 I/F を模擬した。保守 I/F は無線装置の運用データを障害分析サーバ上に構築したテレメトリー基盤及び NW エンジニアリングツールに送信する。NW エンジニアリングツールは遠隔監視や他拠点での監視を想定し、クラウド環境上に試作した。NW エンジニアリングツールには、運用者に対して図 6 のような GUI (Graphical User Interface) を実装した。図 6 は GUI のトップページを示しており、各無線装置の稼働状態及び障害通知結果を運用者に提示する。「稼働状態情報」

一覧に示す無線装置のうち、障害が発生している装置には黄色ハッチングにより運用者に通知する。

次に、障害分析サーバ及び INT スイッチについて説明する。INT スイッチでは **3.**にて説明したテレメトリーデータを生成し、障害分析サーバ上に構築したテレメトリー基盤にテレメトリーデータを送信する。テレメトリーデータ及び無線ネットワーク模擬機能の保守 I/F によって収集した無線装置の運用データを用い、**4.**にて説明したクラスタリング分析を用い、障害要因分析を行う。障害要因分析結果は、NW エンジニアリングツールに送信し、運用者に提示する。

試作した実証システムを用いた障害発生から障害対処フローに関する実証について説明する。図 7

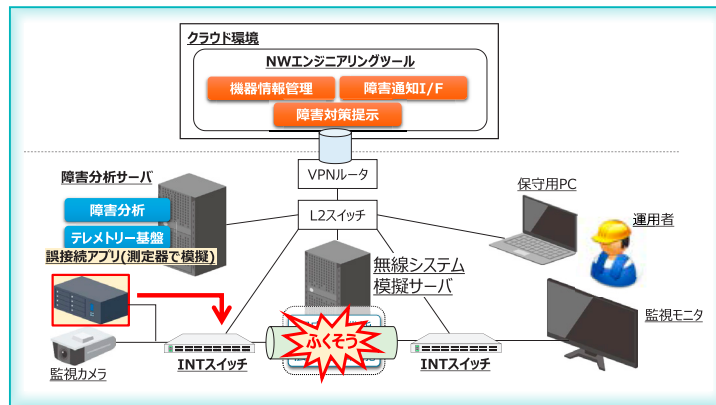


図7 実証シナリオ

に実証シナリオを示す。今回は、アプリケーションのふくそうによる品質劣化を模擬した。具体的には、監視カメラを接続しているネットワークスイッチに、運用者が誤ってトラヒックを発生させるアプリケーションの装置を接続してしまったことを想定する。誤接続アプリケーションのトラヒックは、ネ

ットワークアナライザによってトラヒックを注入することで模擬した。無線システムで伝送できる帯域を超過するトラヒックを注入することにより、無線システムがふくそうし、監視カメラトラヒックに遅延やパケット紛失が発生することで監視映像が乱れる、停止するといった障害が発生する。



① 平常時



② 障害検知時



③ 障害対策案提示



④ 障害対処実行



⑤ 障害復旧

図8 実証中のNWエンジニアリングツールのGUI遷移

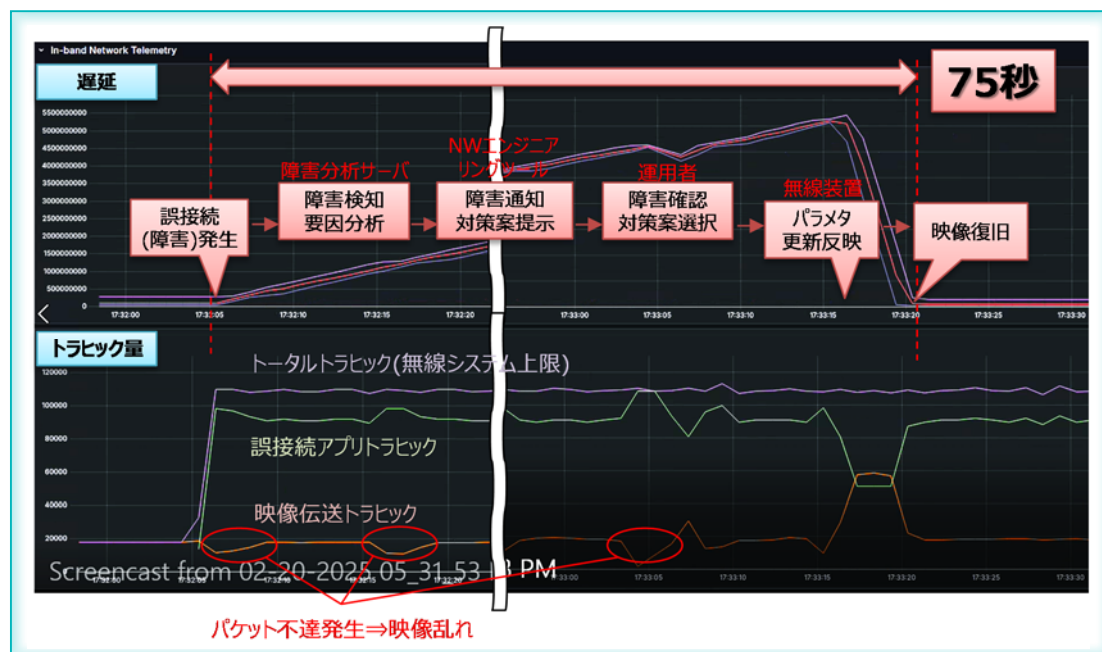


図9 実証中のテレメトリー情報（遅延・帯域）の変化

図8に実証中のNWエンジニアリングツールのGUI遷移を示す。また、図9に実証中のテレメトリー情報の変化を示す。横軸は時間を表しており、上段のグラフはINTスイッチ間の映像トラフィックの遅延時間を示す。下段のグラフは、無線システムを伝送するアプリケーションごとのトラフィック量を示す。

障害発生→障害検知

誤接続アプリケーションを模擬したネットワークアナライザによって、トラフィックを注入すると、図9中の映像トラフィックの遅延が大きくなり、監視カメラアプリケーションとしての障害が発生する（映像停止）。

障害要因分析→障害情報通知

GUIでは図8①→②に遷移し、対象の無線装置に障害が発生したことを、障害分析サーバ内の障害分析機能によって検知し、通知できたことが確認できる。障害要因分析によって、「アプリレイヤでのトラフィックふくそう」が障害要因であることを分析できていることが分かる。

障害情報確認→障害対処方法提示

運用者は、障害通知を選択することで、障害の詳細情報が確認することができる。詳細情報は、障害分析機能が障害検知に使用した根拠となる情報を障害要因分析結果と併せて通知する。NWエンジニアリングツールは、障害要因分析に対応する対策方法を事前にマッピング・管理している。今回は、送信側無線装置から送信するパケットを適切に優先付

けするように、QoS（Quality of Service）パラメータを変更することによって障害に対処する案が運用者に対して提示されている（図8③）。監視カメラトラフィックを高優先伝送するようなQoSパラメータ変更を行うことにより、誤接続アプリケーションのトラフィックの影響を抑制し、本来無線ネットワークシステムで伝送している監視カメラ映像トラフィックに遅延が発生しないようにする。運用者は障害対策案を選択し、実行を判断する。

障害対処実行→障害復旧通知

図8④で示す障害対処実行を行うことで、NWエンジニアリングツールが提示したQoSパラメータ変更を無線システムに反映する。反映後、図9中の遅延時間が小さくなり、障害が復旧したことが分かる。また、図8⑤に示すように障害分析サーバにて障害復旧を検知し、NWエンジニアリングツールにて運用者に対し、障害復旧を通知した。本実証では、ネットワークふくそうという障害に対して、運用者の判断・操作を含め約75秒で障害対処を完了することができた。

6 まとめ

本稿では、無線ネットワークシステムにおける通信障害の対処フローを自動化するために必要となる技術として、運用データを収集するネットワークテレメトリー及び障害要因分析を行うクラスタリング分析について解説した。このような技術を活用することで、無線ネットワークシステムの保守・運用を

部分的に自動化し、省力・省人化を実現することができる。

また、ネットワークテレメトリー・障害要因分析に加え、ネットワーク運用を支援する運用・保守基盤による実証システムを試作し、通信障害発生時の対処フローを実証した結果を紹介した。

■ 文献

- (1) ETSI GS ZSM 003 v1.1.1 (2021-06) Zero-touch network and Service Management (ZSM); End-to-end management and orchestration of network slicing
- (2) A. Giannopoulos, S. Stantideas, N. Kapsalis, P. Gkonis, L. Sarakis, C. Capsalis, M. Vecchio, and P. Trakadas, "Supporting intelligence in disaggregated open radio access networks: architectural principles, AI/ML workflow, and use cases," IEEE Access, vol. 10, pp. 39580–39595, April 2022.
- (3) TM Forum, "TM Forum Introductory Guide Autonomous Networks Technical Architecture," IG1230 Version 1.1.1, Dec. 2022.
- (4) TM Forum, "A white paper of Autonomous Networks: Empowering Digital Transformation For The Telecoms Industry," Release 1.0, May 2019.
- (5) TM Forum(5), C. Kim, A. Sivaraman, N. Katta, A. Bas, A. Dixit, L.J. Wobker, "In-band network telemetry via programmable dataplanes," Proc. ACM SIGCOMM 2015 Conf. Posters and Demos, 2015.
- (6) P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, and D. Walker, "P4: Programming protocol-independent packet processors," ACM SIGCOMM Comput. Commun. Rev., vol. 44, no. 3, pp. 87-95, July 2014.
- (7) In-band network telemetry (INT) dataplane specification v2.1, 2020.
<https://github.com/p4lang/p4- applications/blob/master/docs>
- (8) K. S. Mayer, J. A. Soares, R. P. Pinto, C. E. Rothenberg, D. S. Arantes, and D. A. A. Mellom, "Machine-learning-based soft-failure localization with partial software-defined networking telemetry," J. Opt. Commun. and Netw., July 2021.
- (9) J. Surendiran, B. Mary Nathisiya, A. RadhaKrishnan, J. Mary Stella, M. Marimuthu, and N. Balaji, "Elimination of data modification in sensor nodes of WSN using deep learning model," 2022 Int. Conf. Comput., Power and Commun. (ICCCPC), Dec. 2022.
- (10) L. Anand, Padmalal S, J. Seetha, R. Juliana, PS Naveen Kumar, Gayatri Parasa, "Evaluation of wireless sensor networks module using IoT approach," 2023 Third Int. Conf. Artif. Intell. and Smart Energy (ICAIS), Feb. 2023.
- (11) F. Yang, W. Quan, N. Cheng, Z. Xu, X. Zhang, and D. Gao, "Fast-INT: light-weight and efficient in-band network telemetry in programmable data plane," IEEE 92nd Vehicular Technol. Conf., Nov. 2020.
- (12) J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," Proc. 5th Berkeley Symposium Mathematical Statistics and Probability, vol. 1, pp. 281–297, University of California Press.

(2025 年 8 月 1 日, 10 月 28 日再受付)

三宅裕士 (正員)

2015 東北大学院工学研究科通信工学専攻博士課程後期了。2015 三菱電機株式会社入社。以来、無線ネットワーク制御技術、無線システム運用管理技術の研究開発に従事。博士(工学)。IEEE 会員。



角 武憲 (正員)

2007 同志社大学院工学研究科電気工学専攻修士課程了。2007 三菱電機株式会社入社。以来、無線ネットワーク制御技術、無線共存技術の研究開発に従事。IEEE シニア会員、情報処理学会正会員。



末廣 雄 (正員)

2009 慶大大学院理工学研究科修士課程了。2009 三菱電機株式会社入社。以来、光アクセスシステム、通信ネットワーク運用管理技術の研究開発に従事。

