修 士 論 文

# ホームネットワークにおける<br>トポロジ情報の記述に関する研究

北陸先端科学技術大学院大学
先端科学技術研究科情報

藤巻 伶緒

2018 年 3 月

修 士 論 文

# ホームネットワークにおける トポロジ情報の記述に関する研究


指導教員　丹康雄

審査委員主査　丹康雄
審査委員　篠田陽一
審査委員　リム勇仁
審査委員　Razvan Beuran

北陸先端科学技術大学院大学
先端科学技術研究科 [情報]

1610164 藤巻 伶緒



提出年月: 2018 年 2 月

# 概 要

情報技術の発展に伴い，パーソナルコンピュータ (PC) 以外にも，ホームネットワークに接続可能な機器が増加傾向にある．例えば，情報家電やスマートフォンのようなモバイル端末等が挙げられる．これにより，異なるネットワーク機器や様々な機器が混在することにより，ホームネットワークのトポロジが複雑化するような状況にある．例えば，端末は有線接続されていても，ホームルータとの間に無線や電力線通信区間が入る構成が起こりうる．そのため，ネットワークに関する知識の少ない一般ユーザにとって，複雑なネットワークから自身で障害発生箇所を切り分け，不具合を解決することは極めて困難である．したがって，ホームネットワークの管理し円滑に運用するため，トポロジ情報を自動的に検出し，ユーザや管理者に提供することが必要とされている．

ホームネットワークに接続されたエンド端末やネットワーク機器から，トポロジを検出するための断片的な情報を収集する仕組みが実装されている．その仕組みの一つに，HTIP(Home network Topology Identifying Protocol) がある．HTIP はホームネットワークのトポロジを検出するためのプロトコルである．しかし，この HTIP で収集してきたトポロジ情報の扱い方について学術的な議論はほとんどされていない．他にも，SNMP や LLTD など管理ツールを利用して，ネットワークのトポロジを検出する方法もあるが，何れの方法も複雑なトポロジを形成し，多種多様なデバイスが接続されうるホームネットワークの現状に即していない．また，トポロジに関する情報を収集し，検出されたトポロジを活用するためには，適切な形式でトポロジ情報を記述する必要がある．従来，大規模なネットワークの構築や運用を容易化・自動化するために，ネットワーク記述モデルによるトポロジの情報の記述が行われてきた．しかし，ホームネットワークのトポロジ情報を形式的に記述して取扱うような必然性はまだ高くないとみなされてきており，実例はほとんど存在しない．更に，エンタープライズネットワークやデータセンターのネットワークと比較して，使われている技術の多様性，過去からのシステムの拡張を重ねたことによる見通しの悪さ，全体を把握している設計者や管理者の不在といったホームネットワーク特有の問題もあり，データセンター用の技術を直接転用できない可能性も高い．

本研究では，ホームネットワークにおけるトポロジ情報を統一的に記述する手法を提案する．ホームネットワークでは，異なるネットワークや多種多様な機器が接続されることで，ホームネットワークのトポロジが複雑化している．これに対し，ホームネットワーク

に接続された各機器より収集された管理運用情報からトポロジを自動的に検出し，ネットワーク記述モデルによってトポロジ情報を記述する．

提案手法に基づき，トポロジに関する情報からトポロジを検出するアルゴリズムの実装とトポロジ情報のモデル化を行った．トポロジの検出では，収集された接続構成情報からネットワークトポロジの検出が可能であることを示した．トポロジ情報のモデル化では，検出されたトポロジ情報の記述を可能とし，ネットワークトポロジの把握が可能であることを示した．これにより，アプリケーションの一例としてネットワークトポロジを可視化するアプリケーションを実装した．しかし，システムで利用している接続構成情報はあらかじめ用意した情報であり，HTIP が実装された実際のホームネットワーク環境，或いはシミュレーションされた環境において実験を行う必要がある．

提案手法であるトポロジ情報の記述は，多岐にわたる端末間，或いはネットワーク技術間のトポロジをレイヤ横断的に記述することができ，ホームネットワークのトポロジ全体を把握することができる．また，ネットワーク記述モデルを利用することにより，トポロジ情報をコンピュータで読み込み可能な形式となる．これにより，ホームネットワークの障害原因の特定を容易化・自動化することが可能になる等，トポロジ情報の活用により，ホームネットワークの可用性の向上に貢献できる．

# 目 次

# 図 目 次

# 表 目 次

# 第1章　はじめに

本章では，本研究の研究背景，研究目的，本論文の構成を述べる．

## 1.1　研究背景

近年，情報家電やモバイル端末などが一般家庭に普及し，異なるネットワークや様々な機器が混在することにより，ホームネットワークのトポロジが複雑化するような状況となってきた．例えば，HEMS(Home Energy Management System) では安全性を担保するためコントローラとエネルギー関連機器がホームネットワークに接続された他の機器とは直接 IP 接続できない独立したサブネットで接続されることが多い．また，端末は有線接続されていても，ホームルータとの間に無線や電力線通信区間が入る場合も珍しくない．ネットワークに関する知識の少ない一般ユーザにとって，このような複雑なネットワークから自身で障害発生箇所を切り分け，不具合を解決することは極めて困難である．したがって，ホームネットワークの管理運用を円滑にするため，トポロジ情報を自動的に検出し，ユーザや運用管理者に提供することが必要とされている．そのため，ホームネットワークに接続されたエンド端末やネットワーク機器から，トポロジを検出するための断片的な情報を収集する仕組みが実装されている．その仕組みの一つに，HTIP(Home network Topology Identifying Protocol) がある．HTIP はホームネットワークのトポロジを検出するためのプロトコルである．しかし，この HTIP で収集してきたトポロジ情報の扱い方について学術的な議論はほとんどされていない．他にも，SNMP や LLTD などのプロトコルを利用して，ネットワークのトポロジを検出する方法もあるが，何れの方法も複雑なトポロジを形成し，多種多様なデバイスが接続されうるホームネットワークの現状に即していない．

また，トポロジに関する情報を収集し，検出されたトポロジを活用するためには，適切な形式でトポロジ情報を記述する必要がある．従来，大規模なネットワークの構築や運用を容易化・自動化するために，ネットワーク記述モデルによるトポロジの情報の記述が行われてきた．しかし，ホームネットワークのトポロジ情報を形式的に記述して取扱うよう

な必然性はまだ高くないとみなされてきており，実例はほとんど存在しない．更に，企業ネットワークやデータセンターのネットワークと比較して，使われている技術の多様性，過去からのシステムの拡張を重ねたことによる見通しの悪さ，全体を把握している設計者や管理者の不在といったホームネットワーク特有の問題もあり，データセンター用の技術を直接転用できない可能性も高い．

## 1.2 研究目的

本論文では，ホームネットワークにおけるトポロジ情報を統一的に記述する手法を提案する．ホームネットワークでは，異なるネットワークや多種多様な機器が接続されることで，ホームネットワークのトポロジが複雑化している．これに対し，ホームネットワークに接続された各機器より収集された管理運用情報からトポロジを自動的に検出し，ネットワーク記述モデルによってトポロジ情報を記述する．これにより，多岐にわたる端末間，或いはネットワーク技術間のトポロジをレイヤ横断的に記述することができ，ホームネットワークのトポロジ全体を把握することができる．また，ネットワーク記述モデルを利用することにより，トポロジ情報をコンピュータで読み込み可能な形式となる．これにより，ホームネットワークの障害原因の特定を容易化・自動化することが可能になり，ホームネットワークの可用性の向上に貢献できる．

## 1.3 本論文の構成

本論文は，本章を含めて 7 章で構成される．2 章では，ホームネットワークの環境と管理運用における課題について述べる．また，管理運用においてトポロジ情報の把握の重要性について述べる．3 章では，ネットワークの管理運用技術とネットワーク記述モデルについて説明する．4 章では，ホームネットワークにおけるトポロジの検出手法について提案する．5 章では，提案手法のシステムに基づくシステムの設計と実装について述べる．6 章では，提案手法の有用性について述べる．7 章では，本論文のまとめと今後の課題について述べる．

# 第2章　ホームネットワークの環境と管理運用における課題

本章では，今日のホームネットワークの環境と，ネットワークのトポロジを把握することの重要性について述べる．また，本研究で対象とするネットワーク環境について述べる．

## 2.1　ホームネットワーク環境

図 2.1 に複雑化するホームネットワーク環境の一例を示す．ホームネットワークの環境は，これまでのインフラと異なり，インフラを整備することで接続することができるものではない．ネットワークのケーブルがあり，適切な機器とソフトウェアが提供され，さらに認証などの設定情報を入力しなければ，適切に動作することはない．ネットワークのケーブル事業者，サービス事業者，端末ベンダ，ネットワーク機器のベンダごとにそれぞれ方法が異なり，種類も多いことから全ての相互接続試験を行った上での販売が行われているわけではない．このような環境下で，プロトコルが標準化されており，各機器やサービスは準拠している．相互接続のノウハウも蓄積されてきており，ほとんどが問題なく繋がる状況にある．問題が発生した場合，専門知識を持った保守員が点検を行うが，ネットワークの保守員はネットワークの点検まででアプリケーションの動作を保証することはなく，家電やモバイル機器の保守員はネットワークの点検まではできない．ホームネットワークとアプリケーションの両方の環境の整備はそこに住むユーザが対応せざるをえない状況にある．

エンタープライズのネットワークにおいては，利用者の増加から，自動設定や自動接続，自動認証といったプロトコルが必要とされ，技術開発や運用が進んでいる．ホームネットワークにおいても，同じプロトコルが使われてきており，特に問題がなければ接続できる状況にある．しかし，専門のネットワーク管理者がネットワークの整備と管理をし，配線や配置を把握して，アプリケーションの動作状況を記録しているエンタープライズ環境に対し，ホームネットワーク環境でそのような管理がされていることはほとんどない．し

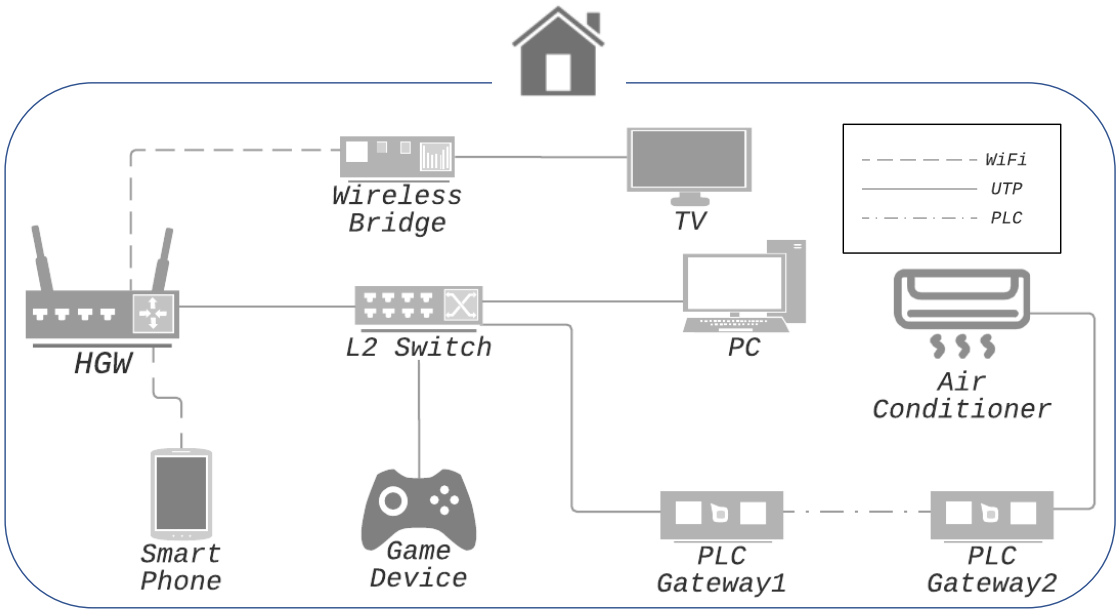


図 2.1: ホームネットワークのトポロジ例

たがって，何か問題が発生した場合に状況を把握したり，対処するといったことが難しくなっている．

### 2.1.1 多種のネットワークの混在

ホームネットワークにおいては，有線のコネクタの数は少なくなく，ネットワークケーブルの配線の取り回しが簡単でないことから無線 LAN が普及している．モバイル機器は単独で使えることが利便性の向上につながっており，インタフェースが無線 LAN や Zigbee，Bluetooth 等の無線しかないものも多く存在する．無線は便利であるが，目に見えないため管理が難しいという面がある．最初に接続する際には，セキュリティ上の問題からパスワードを要求することも多い．

### 2.1.2 多種多様な機器の接続

エンタープライズのネットワークと比較して，ホームネットワークに接続される機器の種類は多い．PC やプリンタだけでなく，家電，ゲーム機，スマートフォン，タブレット端末等多岐にわたる．また，アプリケーションは，メールや Web 閲覧にとどまらず，ファー

ムウェアのアップデートなどのシステム管理の用途のものがある．従来，家電は一度販売してからのアップデートがないのが一般的であったが，今日では新機能や不具合の修正のために，ネットワークなどでのアップデート機能を取り入れるものも増えている．

### 2.1.3　運用管理者の不在

ホームネットワークは，特定の管理者はいない状態で運用されている場合が多い．特別な操作をしなくても接続が可能になる開発され普及した結果，特にネットワークに関する知識がなくても利用することができる．しかし，どうやってどうやって動いているかを把握している人がいないため，もし何か問題が発生し場合に適切に対応するのは難しい状況にある．

## 2.2　ネットワークトポロジ把握の重要性

ネットワークのトポロジ情報は，さまざまな状況で価値がある．トポロジ情報は，ネットワークの管理において，障害の検出と回避，ネットワークインベントリ，性能予測，ネットワークシミュレーション等での利用が考えれる．また，ネットワークセキュリティの観点から，トポロジ情報は，脅威検出，ネットワーク監視，ネットワークアクセス制御等での利用が考えられる．ホームネットワークの全体を見通す管理者の不在と多種多様な機器が混在し構成が複雑化している現状では，手動でのネットワークマッピングが困難になってきている．したがって，自動でトポロジを検出し，トポロジ情報を記述することは，ホームネットワークの管理において非常に重要な役割を果たす．

物理的トポロジーの検出に関する研究は、ネットワーク要素がインテリジェントであり、トポロジー関連情報の問い合わせが可能であると想定される協調的ネットワーク環境に主に焦点を当てている．

### 2.2.1　ネットワークの管理

ネットワーク管理者或はユーザーは，多くの場合，障害検出と回避を実行する必要のあるネットワークの問題に直面している． ネットワークの問題をトラブルシューティングするために，ネットワークのトポロジマップを使用して問題の領域を切り分けすること

ができる．トポロジマップは，インフラストラクチャの脆弱性を特定するのにも役立ち，ネットワークは，より多くの冗長性を提供するように適合させることが 可能になる．

ネットワーク管理から，ネットワークトポロジ情報をネットワーク管理に適用することができる． ネットワークトポロジ情報は，新しい機器を追加する場所を決定し，現在のハードウェアが正しく構成されているかどうかを判断する際に役立てることができる．また，ネットワーク管理者或はユーザは，ネットワークのボトルネックや障害を見つけることが可能になる．

Network Management System(NMS) は，トポロジ情報も使用してネットワーク管理を支援している．代表的なシステムには，IBM の Tivoli1，Hewlett-Packard の OpenView2 やオープンソースの Open-NMS3 等がある．

### 2.2.2 性能予測

性能予測では，トポロジ情報を使用してネットワーク認識のアプリケーションのパフォーマンスを最適化し，分散アプリケーションのパフォーマンスを最適化することができる．トポロジの情報は，特定のネットワーク上で，特定のサービス品質 (QoS) を提供するかどうかの判断するのに役立てることができる．例えば，ネットワークが VoIP(Voice over IP) などのマルチメディア技術をサポートするかどうかを判断するには，ネットワークトポロジの情報が不可欠になる．

### 2.2.3 シミュレーション

アプリケーション領域であるネットワークシミュレーションの精度は，現実的で正確なネットワークトポロジに依存する．シミュレーションに使用される生成されたトポロジ，必ずしも現実のトポロジーと一致するとは限らない．現実のネットワークトポロジーを正確に測定する必要が生じる．ネットワークシミュレーションは，研究者がネットワークの現在の動作を理解するだけでなく，将来のネットワーク変更の影響を理解するのに役立てることができる．境界防御の機構である侵入検知システム (IDS:Intrusion Detection System) は，ネットワークトポロジ情報を考慮することで恩恵を受ける．IDS が正しく配置されていないと，誤検知などの問題が発生する可能性がある．

### 2.2.4　セキュリティ

トポロジ情報には，セキュリティ上のいくつかのアプリケーションで利用される可能性がある．従来，ファイアウォールは外部からの脅威から保護するためにネットワークのエッジに配置がされていた．ネットワークに対する内部からの脅威も一般的になりつつある．したがって，ファイアウォールの配置とネットワークのセキュリティポリシー管理は，ネットワークトポロジの影響を受ける．

ファイアウォールや侵入検知システムの問題は，ネットワークアクセス制御 (NAC:Network Access Control) の分野の研究で関心がある．セキュリティポリシーに準拠していない機器は，スイッチポートを物理的に無効することによって，ネットワークインフラストラクチャへのアクセスを拒否することができる．しかし，ネットワークのトポロジ情報と接続された機器に関する知識が不足すると，NAC ソリューションの有効性が大幅に低下する可能性がある．

## 2.3　対象とするネットワーク環境

本研究では，一つのリンクレイヤブロードキャストドメインを研究の対象とする．図 2.2 に示すように，一つのリンクレイヤブロードキャストドメイン外の Zigbee や Bluetooth のような非 IP ネットワークや，ルータまたは L3 スイッチで分離された IP ネットワークについては対象外とする．また，ホームゲートウェイ (HGW) の FDB にはホームネットワークの WAN に当たる情報を含んでいないこととする．

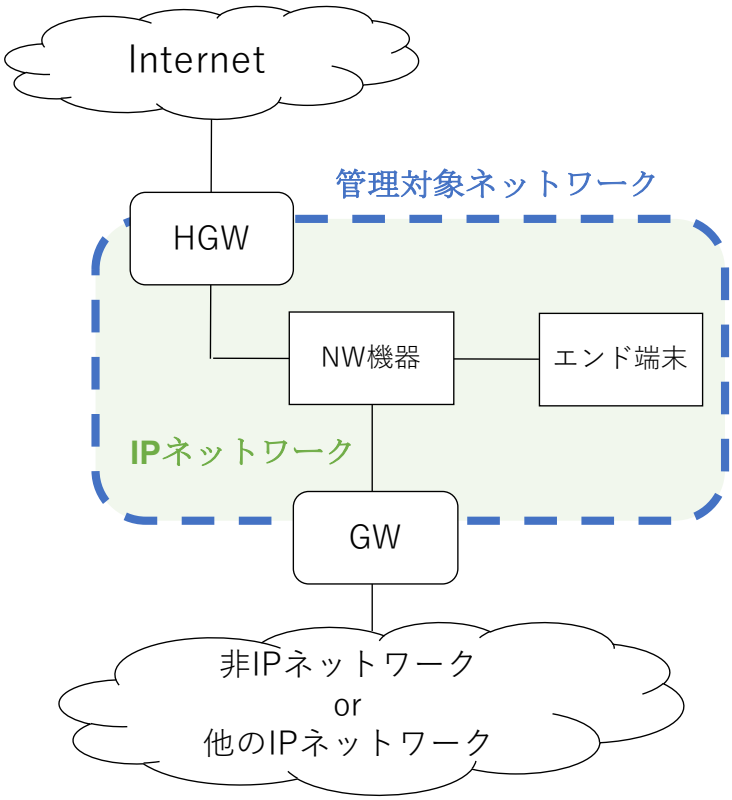


図 2.2: 対象のネットワーク環境

# 第3章　関連研究

本章では，まず本研究の背景となるネットワークの管理に関する主な技術・規格について説明する．ホームネットワークに限らず一般的に使われる技術からホームネットワークを対象とした管理技術まで説明する．次に，ネットワーク記述モデルについて提案手法と関連する技術を説明する．

## 3.1　ネットワーク管理運用技術

ネットワークは大規模化・複雑化しており，これによりネットワーク運用のリスクが高まっている．ネットワーク管理のために，いくつかの技術・規格が存在しており，これらの技術・規格により適切な情報の取得を行うことができる．この節では，代表的なネットワーク管理運用ツールやネットワーク管理運用のための規格について説明する．

### 3.1.1　SNMP(Simple Network Management Protocol)

SNMP(Simple Network Management Protocol)[1] は，IP ネットワーク上の機器を監視・制御するためのプロトコルである．図 3.1 に示すように，SNMP では Manager，Agent，SNMP プロトコル，管理情報の四つで構成される．Manager はネットワーク管理者が利用するアプリケーションで，管理対象に命令を行い，管理対象からの応答の蓄積や分析をする．Agent は管理対象上のアプリケーションで，Manager からの要求に対して応える役割をする．管理情報は，Agent が管理する機器に関する情報であり，管理情報データベースとして機器に保存されている．SNMP プロトコルは，Manager と Agent 間で通信を行うために使われるプロトコルである．


図 3.1: SNMP の基本となる構成要素

### 管理情報

Agent は Manager からの問い合わせに応じ機器の状態を調べる際に，機器が保持している管理情報のデータベースにアクセスする．管理情報のデータベースは MIB(Management Information Base)[2] に従って定義されており，管理対象オブジェクトから構成される．管理対象オブジェクトは，SMI(Structure of Management Information)[3] によってその構造と使用可能なデータ型が定義され，オブジェクト識別子とその値から構成される．SMI で管理対象オブジェクトとオブジェクト配列を定義する公式のモデルとして，抽象構文表記法 1(ASN.1: Abstract Syntax Notation One) が使用され，エンコード方式には BER(Basic Encoding Rules) が利用されている [4].

### SNMP プロトコル

Manager と Agent では SNMP プロトコルを利用して，機器の管理情報の取得や設定をする．SNMP には，SNMPv1[1]，SNMPv2，SNMPv2c，SNMPv3 の四つのバージョンがあるが，本論文では SNMPv1 について簡単に説明する．SNMPv1 メッセージパケットの構造を図 3.2 に示す.

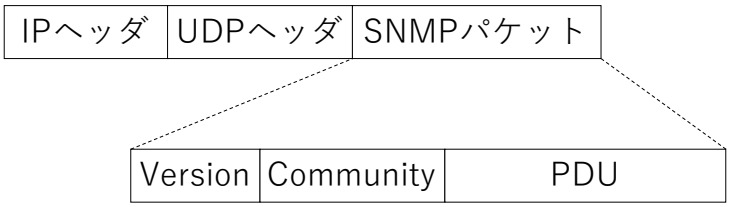


図 3.2: SNMPv1 のメッセージパケットのフォーマット

SNMP プロトコルではネットワーク層に IP，トランスポート層に UDP を利用している．SNMP パケットは，使用している SNMP のバージョンを示す Version フィールド，Manager

と Agent 間の認証に使用される Community Name フィールド，管理情報の取得や変更などのオペレーションのコマンドを含む PDU(Protocol Data Unit) フィールドで構成される．Community Name とは，Manager と Agent 間のパスワードであり，Read-only と，Read-write，trap の 3 種類がある．PDU には，SNMPv1 の場合，GetRequest，GetNextRequest，GetResponse，SetRequest，trap の五つがある．SNMPv1 における PDU の一覧を表 3.1 に示す．

表 3.1: SNMPv1 における PDU の一覧

| PDU | 送信側 | 用途 |
|---|---|---|
| Get Request | Manager | Agent から取得したい情報をオブジェクト ID を指定して要求 |
| GetNext Request | Manager | 直前に指定したオブジェクト ID の次のオブジェクト ID を指定して要求 |
| Set Request | Manager | Agent の設定を変更したい場合，オブジェクト ID を指定して要求 |
| Get Response | Agent | Manager から要求されたオブジェクト ID に対応して情報を返信 |
| trap | Agent | Agent の機器状態に変化があった場合，自発的に送信 |

SNMP は，主にエンタープライズのネットワーク機器に搭載されている．現状，ホームネットワークに接続される家庭向けの安価なネットワーク機器の多くが，パケット・フレームの転送機能だけを持っており，SNMP を搭載していることはない．それらのネットワーク機器が，概して処理性能が低く，大量の MIB 情報を保持することが難しいためである．また，家庭向けの安価なネットワーク機器では，L2 の処理のみが行われ，SNMP 等の L3 プロトコルの処理は不可能である．

### 3.1.2 LLTD(Link Layer Topology Discovery)

LLTD(Link Layer Topology Discovery)[5] は，ネットワークのトポロジを検出し，サービス品質の診断を行うするデータリンク層のプロトコルである．Windows Relly 技術の一環として米マイクロソフトにより開発され，ライセンスされている．LLTD にはトポロジを描画する Mapper と管理対象の機器となる Resonder の二つで構成される．ユーザがネットワークマップを起動すると，Mapper は探索フレームをブロードキャストで送信し，Responder が管理情報を返す仕組みとなっている．Mapper と Responder でやり取りされる情報は，機器のデバイス名，IP アドレス，無線 LAN 情報 (通信規格，利用中の

SSID/BSSID)，管理 URL 等がある．また，Mapper は Responder に対して，特定ノードとのテスト通信を行い，テスト結果を収集，分析することでトポロジを描画する．LLTD では，米マイクロソフトによりライセンスされていることから，一般的なホームネットワークで利用する場合，汎用性に欠けるといった問題がある．

### 3.1.3 LLDP(Link Layer Discovery Protocol)

LLDP(Link Layer Discovery Protocol)[6] は，機器の種類や設定情報などを近隣の機器に通知するデータリンク層のプロトコルであり，IEEE802.1AB で標準化されている．LLDP に対応した機器は，自身の管理情報を定期的 (IEEE802.1AB 規格では 30 秒を推奨) にマルチキャスト・アドレス宛に送信する．LLDP では管理情報を任意で追加することが可能であり，扱う項目として基本項目 (必須項目・オプション項目) と拡張項目がある．基本項目を表 3.2，拡張項目を表 3.3 に示す．基本項目には，必須項目が三つ，オプション項目が五つ規定されている．拡張項目では，IEEE802.3(LAN) と IEEE802.1(VLAN) 関連の項目が規定されてる．拡張項目については IEEE802.1AB の規格に準拠していれば，業界団体やベンダーが独自に定義することが可能であり，HTIP ではこの拡張項目を利用している．

表 3.2: LLDP 基本項目一覧

| 項目 (必須) | 用途 |
|---|---|
| Chassis ID | 機器の識別情報 (Mac アドレス等) |
| Port ID | 機器が LLDP を送信したインターフェース情報 |
| Time To Live | 情報の有効期間 |
| 項目 (オプション) | 用途 |
| Port Description | インタフェースの概要 |
| System Name | システム名 |
| System Description | 機器の名称やバージョン |
| System Capabilities | 機器の種別 |
| Management Address | 管理用アドレス |

表 3.3: LLDP 拡張項目一覧

| 項目 (IEEE802.3 関連項目) | 用途 |
|---|---|
| MAC/PHY Configuration Status | オート・ネゴシエーションのサポート/利用状況，使用しているインタフェースの種類等 |
| Power Via MDI | PoE(Power over Ethernet) などの電源供給に対するサポート/利用状況，ポートの給電クラス等 |
| Link Aggregation | リンク・アグリゲーションのサポート/利用状況 |
| Maximum Frame Size | フレームサイズの最大値 |
| 項目 (IEEE802.1 関連項目) | 用途 |
| Port VLAN ID | ポート VLAN の ID |
| Port and Protocol VLAN ID | ポート/プロトコル VLAN のサポート/利用状況，VLAN ID |
| VLAN Name | VLAN 名と，それに割当てられた ID |
| Protocol Identity | 利用可能なプロトコル |

### 3.1.4 UPnP(Universal Plug and Play)

UPnP(Universal Plug and Play)[7] は，UPnP フォーラムが定めた，デバイスを接続しただけでネットワークに参加することを可能にするためのプロトコルの集合である．UPnP はネットワーク上での汎用的な通信方法のみを規定する UPnP デバイス・アーキテクチャ (UPnP DA) と，その上位層にある UPnP デバイス制御プロトコル (UPnP DCP) に分別される．UPnP DA の仕様は，大きく分けて Addressing，Discovery，Description，Control，Eventing，Presentation の六つで構成される．UPnP DA の仕様を表 3.4 に示す．UPnP では，ネットワークに接続されたデバイスと，そのデバイスに対して制御する機構であるコントロール・ポイントが定義されている．HTIP では，Description においてデバイス内容が XML 形式で記述されている DDD(Device Description Document) 中の拡張を行い利用している．

表 3.4: UPnP 仕様・機能

| 仕様 | 機能 |
|---|---|
| Addressing | デバイスの IP アドレスを決定 |
| Discovery | ネットワーク上のデバイスを検出 |
| Description | 検出デバイスの内容を取得 |
| Control | デバイスの制御 |
| Eventing | デバイスの状態を監視 |
| Presentation | WEB からのデバイス状態の確認や制御 |

### 3.1.5 ITU-T G.9973(HTIP)

HTIP[8] は，ホームネットワークのトポロジを検出するための情報を収集するプロトコルである．また，リンクレイヤブロードキャストドメインにおいてのみ有効である．HTIP において，接続される HTIP-エンド端末には UPnP Device Architecture の Controlled Device が実装 (L3Agent)，または LLDP Agent(Transmit only) が実装 (L2Agent) され，HTIP-NW 機器には LLDP Agent(L2Agent) が実装されている必要がある．また，Manager が各 Agent から機器情報及び接続構成情報を収集することでホームネットワークのトポロジを検出する．Manager は，ホームネットワーク内の任意の端末に存在することを想定している．機器情報は，各 Agent 毎に管理されており，少なくとも以下の四つの情報から構成される．その他に，障害発生箇所の切り分けに有用な情報として，チャネル使用状態情報，電波強度情報，ステータス情報等を通知することが可能である．

(a) 区分

- 機器の種別を表し，例えば ”TV ”や ”PC ”等の種別を表す値．

(b) 型番

- メーカ毎に付与される機器の型番．

(c) メーカーコード

- 機器を製造した会社名を表す値．これは，IEEE に登録されたカンパニー ID(OUI コード) で記述．

(d) 機種名

– メーカ毎に付与される機器のブランド名やシリーズ名を表す値.

接続構成情報は，NW 機器が保持する情報である Forwarding Data Base(FDB) と同義である．NW 機器におけるポートと，そのポートに接続されたエンド端末の MAC アドレス，或いは他の NW 機器の MAC アドレスが対になった情報である．なお，この規格では L2Agent と L3Agent が情報を送信するプロトコルについて規定されており，Manager の機能については詳しく記述されていない.

### L2 Agent

L2 Agent は，NW 機器またはエンド端末に常駐し，Manager に対して LLDP を利用し，機器情報と接続構成情報を通知する必要がある．L2Agent は少なくとも，HTIP-NW 機器を区別可能な Chassis ID と，接続構成情報，この HTIP-NW 機器自身の MAC アドレスリスト，機器情報を管理オブジェクトとして保持する．L2 Agent が送信する LLDPDU のフレーム構成を図 3.3 に示す．Destination MAC Address から LLDP Ethertype はイーサネットヘッダである．L2 Agent は，イーサネットヘッダの送信元 MAC アドレスを IEEE802.3 で定められたブロードキャストアドレスである FF-FF-FF-FF-FF-FF に設定する．イーサネットヘッダの送信元 MAC アドレスは，そのポートの MAC アドレスとなる．LLDPDU フレームを受信した NW 機器は，IEEE802.1D を参照し，受信したフレームの送信先を MAC アドレスのアドレスグループに応じた挙動をする必要がある.


図 3.3: LLDPDU フレーム構成

L2 Agent は，IEEE802.1AB で規定されているベンダー拡張フィールド (IEEE802.1AB において TLV Type = 128) を利用し，機器情報と接続構成情報を TLV として送信する．ベンダー拡張フィールドを図 3.4 に示す．この TLV は，TTC の OUI コード E0-27-1A と，TTC で規定した情報を格納する．また，TLV における文字列の長さは，オクテット単位で表記をする必要がある.

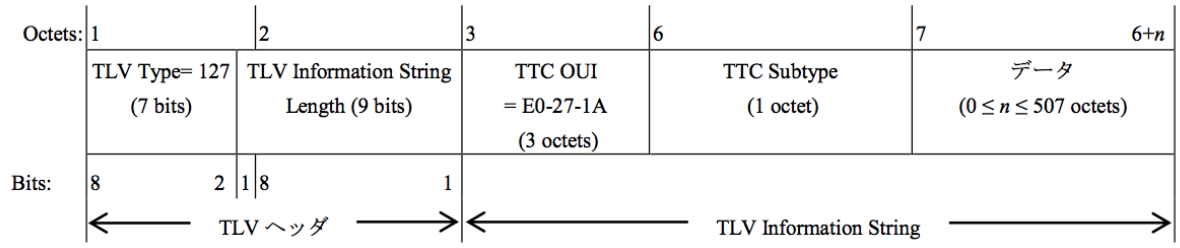


図 3.4: ベンダー拡張フィールドの利用法

LLDPDU には，IEEE802.1AB で実装必須となっている TLV を格納する．また，HTIP-NW 機器上の LLDP Agent から送信される一つの LLDPDU には，(a) 区分，(b) 型番，(c) メーカーコード，(d) 機種名，機器を特定する ID が格納された TLV と，接続構成情報の TLV がそれぞれが含まれている必要がある．図 3.4 における TTC Subtype を表 3.5 に示す．Subtype が 1 の時，TLV は機器情報を示し，Subtype が 2 の時は接続構成情報を示す．

表 3.5: TTC Subtype と格納する情報の対応

| TTC Subtype | データ内容 | 実装 |
|---|---|---|
| 1 | 機器情報 | 実装必須 |
| 2 | 接続構成情報 | HTIP-NW 機器：実装必須<br>HTIP-エンド端末：実装不要 |
| 3 | MAC アドレスリスト | オプション実装 |
| 4 | 拡張接続構成情報 | HTIP-NW 機器：オプション実装<br>HTIP-エンド端末：実装不要 |
| 5 | 拡張 MAC アドレスリスト | HTIP-NW 機器：オプション実装<br>HTIP-エンド端末：実装不要 |
| 6 | 設定情報 | オプション実装 |
| 0, 7-255 | 予約領域 | |

### L3 Agent

L3 Agent は，HTIP-エンド端末に常駐し，ホームネットワーク内の任意の端末上に存在する Manager に対して UPnP controlled device の機能を利用し，機器情報を通知する．機器情報の通知には，DDD の root device の Basic Device Information 部を利用する．DDD

に対して新規エレメントの追加，既存エレメントへの値の記述をする．機器情報の各項目と，その機器情報を格納するエレメント，またエレメントに格納可能な文字列の最大長，送信のタイミング及び送信方向の対応を表 3.6 に示す．

表 3.6: 機器情報とそれを格納する DDD エレメント対応

| 機器情報を格納するエレメント | 格納される機器情報 | エレメントに格納可能な最大長 (octets) | 送信タイミング | 送信方向 |
|---|---|---|---|---|
| htip:X_DeviceCategory | 区分 | 255 | 定期 (LLDPDU 送信間隔) | L3 Agent → Manager |
| htip:X_ManufacturerOUI | メーカーコード | 6 | 定期 (LLDPDU 送信間隔) | L3 Agent → Manager |
| modelName | 機種名 | 31 | 定期 (LLDPDU 送信間隔) | L3 Agent → Manager |
| modelNumber | 型番 | 31 | 定期 (LLDPDU 送信間隔) | L3 Agent → Manager |
| htip:X_ChannelStatus | チャネル使用状態情報 | 3 | 定期 (LLDPDU 送信間隔) | L3 Agent → Manager |
| htip:X_Rssi | 電波強度情報 | 3 | 定期 (LLDPDU 送信間隔) | L3 Agent → Manager |
| htip:X_ErrorRate | 通信エラー率 | 3 | 定期 (LLDPDU 送信間隔) | L3 Agent → Manager |
| htip:X_Status | ステータス情報 | 64 | 定期 (LLDPDU 送信間隔) | L3 Agent → Manager |
| htip:X_RT | 応答時間 | 6 | 定期 (LLDPDU 送信間隔) | L3 Agent → Manager |
| htip:X_NumAss | 関連デバイス数 | 3 | 定期 (LLDPDU 送信間隔) | L3 Agent → Manager |
| htip:X_NumAct | アクティブノード数 | 3 | 定期 (LLDPDU 送信間隔) | L3 Agent → Manager |
| htip:X_LQ | 無線品質 | 3 | 定期 (LLDPDU | L3 Agent → |

表は次ページに続く

表 3.6: 機器情報とそれを格納する DDD エレメント対応

| 機器情報を格納する<br>エレメント | 格納される<br>機器情報 | エレメントに<br>格納可能な<br>最大長 (octets) | 送信<br>タイミング | 送信方向 |
|---|---|---|---|---|
| | | | 送信間隔) | Manager |
| htip:X_RetC | 再送数 | 3 | 定期 (LLDPDU<br>送信間隔) | L3 Agent →<br>Manager |
| htip:X_CPU | CPU 使用率 | 3 | 定期 (LLDPDU<br>送信間隔) | L3 Agent →<br>Manager |
| htip:X_MEM | メモリ使用率 | 3 | 定期 (LLDPDU<br>送信間隔) | L3 Agent →<br>Manager |
| htip:X_HDD | HDD 使用率 | 3 | 定期 (LLDPDU<br>送信間隔) | L3 Agent →<br>Manager |
| htip:X_Power | バッテリ残量 | 3 | 定期 (LLDPDU<br>送信間隔) | L3 Agent →<br>Manager |
| htip:X_ComInterval | 通信品質関連情報の，サンプリング間隔，送信間隔 | 64 | 不定期 (設定時のみ) | L3 Agent →<br>Manager(設定)<br>L3 Agent →<br>Manager(応答) |
| htip:X_SysInterval | 端末品質関連情報の，サンプリング間隔，送信間隔 | 64 | 不定期 (設定時のみ) | L3 Agent →<br>Manager(設定)<br>L3 Agent →<br>Manager(応答) |

## 3.2 ネットワーク記述モデル

情報を構造化することでシステムにおけるデータの交換を可能とし，利便性を向上させることを情報のモデル化という．ホームネットワークのトポロジ情報のマルチベンダー間での相互運用性の可能性を高めるためにネットワークトポロジのモデル化が不可欠である．本節では，既存のネットワーク記述モデルのうち，トポロジ情報の記述に特化した関連技術として，Physical Topology MIB，NDLs，YANG data model for Network Topology，NetJSON について述べる．

### 3.2.1 Physical Topology MIB

Physical Topology MIB[9] は，ネットワークの物理トポロジとそのトポロジ内の関連システムを記述することを目的に開発された．Physical Topology MIB では，マルチベンダーの相互運用性を促進し，標準的な管理ツールを使用してネットワーク管理の物理トポロジ情報を発見し再利用を可能にする．また，ネットワーク管理において不整合や誤作動による特定の構成を発見し，上位層での通信が損なわれる可能性を抑える．さらに，特定された構成の不一致や誤作動を修正するためのリソースの変更や再構成を行う際，ネットワーク管理を支援するための情報を提供する．

### 3.2.2 NDLs(Network Description Languages)

記述言語は，一般に，システム及びその特性の高水準での記述を可能にするツール (形式言語) である．これらの言語は，記述されたシステムの計画，分析，及びリソースの詳細把握をサポートする．記述言語の出力はほとんどの場合，解析ソフトウェアで入力として使用できる XML となる．

#### NDL(Network Description Language)

NDL(Network Description Language)[10] は，オランダの Amsterdam 大学の研究グループ (SNE) によって提案したネットワークリソースの記述方式である．NDL は，ネットワークを記述するためのスキーマを RDF(Resource Description Framework) を利用して定義している．NDL はパケット交換と回線交換が混在するようなハイブリットネットワークにおける運用コストの削減を目的としている．ネットワークの構成要素として Device，

Interface，Link を定義し，その関係性を　RDF で記述することで分散した情報の集約やトポロジの自動生成による運用支援を可能としている．図 3.5 は NDL のクラスとプロパティの概要である．

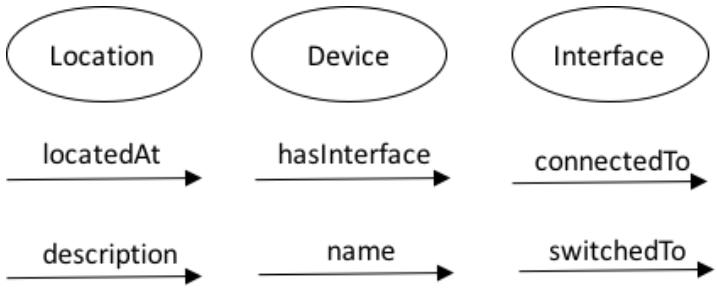


図 3.5: NDL のクラスとプロパティ概要

### NML(Network Markup Language)

NML(Network Markup Language)[11] は，Open Grid Forum によって定義され，コンピュータネットワークとリソースの接続を記述するための標準的なオントロジである．NML のネットワーク記述は，XML と RDF で表現される．NML では，異なるレイヤのトポロジ，及びレイヤとネットワークのやり取り方法を記述することを可能としている．これにより，ネットワークにおける経路計算のためのトポロジ情報の交換等をすることを可能とする．また，NML は障害検出や可視化のための重要なツールとなり得る．

### VXDL(Virtual Resources and Interconnection Network Description Language)

VXDL(Virtual Resources and Interconnection Network Description Language)[12] は，グリッドコンピューティングを対象としている．グリッドの実装には特殊な操作特性があり，グリット環境で実行中のアプリケーションを妨げることなく，関係するリソースを動的に再割り当てすることが必要になる．これらのリソースには，ノードとして機能する仮想マシン，相互接続，ネットワーク帯域幅などのタイプがある．VXDL は，仮想インフラストラクチャのリソース，ネットワークトポロジ，それらの特定の属性の記述を可能にする．

### INDL(Infrastructure and Network Description Language)

INDL(Infrastructure and Network Description Language)[13] は，テクノロジーに依存しない，コンピューティングインフラストラクチャーを記述することを目的としている．記述には接続される物理リソースとネットワークインフラストラクチャが含まれる．また，リソースの仮想化と，それらのリソースによって提供されるサービスを記述するために必要な語彙を提供している．さらに，エネルギー消費などのリソースの動作面を記述するためなどに容易に拡張することが可能である．INDL は，EU-FP7 という欧州の研究プロジェクトにおける NOVI Project[14] や GEYSERS Project[15] において研究開発が進められている．

INDL は，図 3.6 のように抽象クラスと，それの関連及びプロパティの階層を定義するオントロジーに基づいている．Resource と Service の二つの主要なクラスは，ネットワーク設定の中核となる構成要素と特性を記述するために使用される．さらに，Node，NodeComponent，NetworkElement の三つのサブクラスは，物理/仮想ノード，それらの特性 (CPU やメモリ等) 及びネットワーク接続をそれぞれモデル化することを可能とする．仮想化は，Node のサブクラスである VirtualNode を介してモデル化される．サービスの概念は，サービスの所有者或いはプロバイダのいずれかとみなされる参加ノードによって抽象化される．

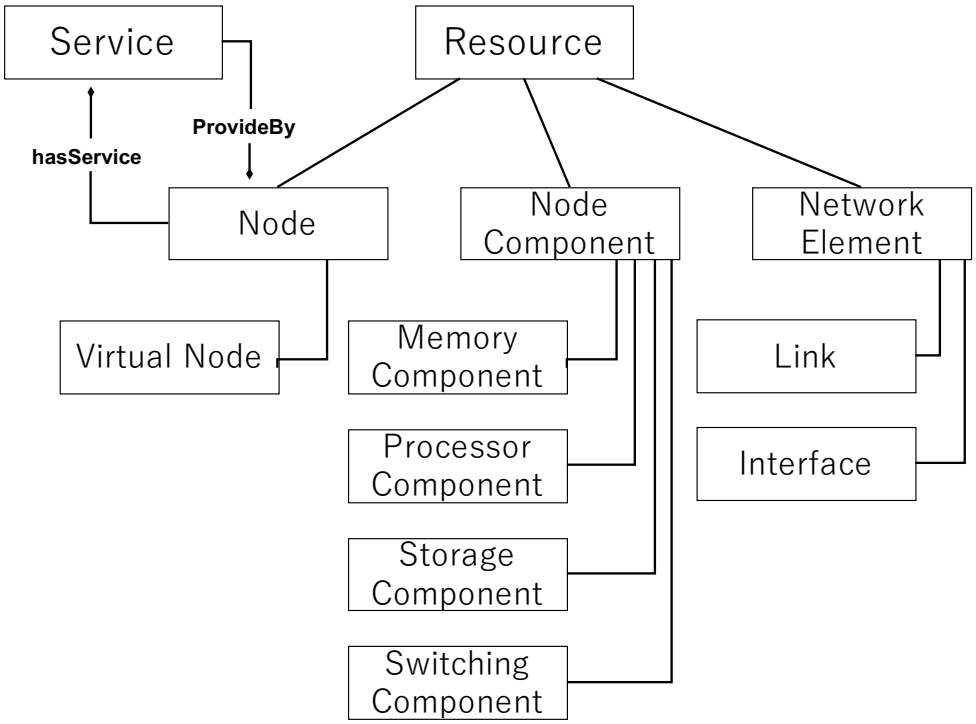


図 3.6: INDL のクラス階層

### 3.2.3 YANG data model for Network Topology

YANG data model for Network Topology[16] とは，YANG(Yet Another Next Generation) というデータモデルでネットワークトポロジを記述することを目的としたネットワーク記述モデルである．YANG[17][18] は，ネットワーク管理機器の管理プロトコルである NETCONF(Network Configuration Protocol) で使用される設定項目や状態などの構造を人が解釈しやすい形式で記述することを目的としたデータモデル言語である．

### 3.2.4 NetJSON

NetJSON[19] とは，L2 および L3 ネットワークの基本的な要素を記述するために設計された JSON(JavaScript Object Notation) に基づくネットワーク記述モデルである．JSON[20] は，軽量なデータ記述言語の一つである．構文は JavaScript におけるオブジェクトの表記法を基本としているが，JavaScript 専用のデータ形式ではなく，様々なソフトウェアやプログラミング言語間のデータの交換に使えるよう設計されている．NetJSON では，

NetworkRoutes，DeviceConfiguration，DeviceMonitoring，NetworkCollection，NetworkGraph の五つのデータ記述モデルが定義されている．

- NetworkRoutes
- DeviceConfiguration
- DeviceMonitoring
- NetworkCollection
- NetworkGraph

このうち，NetworkGraph は，ネットワークのノードとリンクをリストとして定義しており，ネットワークトポロジの可視化を目的として開発された記述モデルである．

# 第4章　提案手法

ネットワークのトポロジ情報の記述とは，ネットワークに接続された各機器から集められたトポロジ情報を，形式的な情報として記述を行うことである．ネットワークに接続された機器からトポロジに関連する情報を収集する方法が存在する．また，いくつかのネットワーク記述モデルが存在する．

前章では，トポロジ情報の収集に，ネットワーク管理技術として SNMP や LLTD は，ホームネットワーク上で実装することは難しいことを示した．また，ホームネットワークトポロジの特定を目的に開発された HTIP においても，Manager に集められた情報をどのように処理するかは規定されていないことを指摘した．

本章では，そのようなホームネットワークにおいてトポロジ情報を記述するためにはどのような機能が必要か提案し議論する．

## 4.1　トポロジ情報の記述の有用性

本論文で述べるトポロジ情報の記述とは， 自動的に検出されたトポロジ情報を統一的に記述することである．つまり，トポロジ情報を自動的に検出する方法と，その検出されたトポロジ情報を記述する手法が必要となる．

トポロジを自動的に検出し，形式的にトポロジ情報を記述することで，トポロジ情報の利用性を向上させる．ネットワークの不具合の発生時には，ネットワーク管理者或はユーザに対して，有用な情報を提供することが可能になる．そのため，トポロジ情報の記述はネットワークを管理して運用していく上で有用であると言える．

また，前章で述べたネットワーク管理運用技術を使用することで，ネットワークに関するトポロジ関連の情報を収集することが可能である．しかし，これらの手法は，トポロジ情報を収集する方法のみにとどまり，トポロジを検出を行っているものはなく，ホームネットワークにおけるトポロジ情報を統一的に記述する手法は存在しない．

## 4.2　ネットワークトポロジ検出手法

ネットワーク運用管理技術を使用して，収集されたトポロジ関連の情報からトポロジ情報を把握するには，トポロジの検出を行う必要がある．本節では，トポロジ検出アルゴリズムについて述べる．

### 4.2.1　前提条件

NW 機器の全 FDB を一つの機器 (以下，Manager) で収集し，この Manager にてネットワークのトポロジを検出することを想定している．一般的には HGW が Manager として動作することが多いため，以下においても HGW を Manager とする．Manager は，HTIP を用いて FDB を収集し，リンクレイヤのトポロジを検出する．また，HTIP では NW 機器は FDB が含まれた LLDP フレームを定期的にブロードキャストする．

#### グラフ表現

ネットワークは，各機器をノード，接続媒体をエッジで表現することでグラフとして図 4.1 のように表現することができる．図 4.1 は図 2.1 のネットワークのグラフ表現である．グラフの丸の中のラベルはその端末の MAC アドレスを示している．より簡易化するため，これらのラベルはノードの名前とする．例えば，図 4.1 内のノード D は図 2.1 内の PC であり，PC の MAC アドレスは D である．また，NW 機器については灰色の丸で表現されており，Wireless Bridge を WR，L2 Switch を L2SW，PLC Gateway1 を PLC1 として，名前を省略している．NW 機器から伸びるエッジに記載された数字はポート番号を示している．

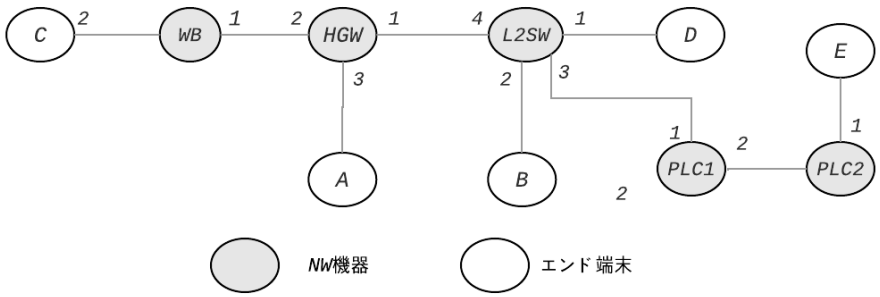


図 4.1: 図 2.1 のグラフ表現

**FDB**

表 4.1 は，図 4.1 の NW 機器の FDB を示している．なお，一つの FDB は一つの NW 機器における FDB である．

表 4.1: 図 4.1 のネットワークに対応した FDB

| HGW | |
|---|---|
| port | MAC Address |
| 1 | L2SW, PLC1, PLC2, B, D, E |
| 2 | WB, C |
| 3 | A |

| L2SW | |
|---|---|
| port | MAC Address |
| 1 | D |
| 2 | B |
| 3 | PLC1, PLC2, E |
| 4 | HGW, WB, A, C |

| WB | |
|---|---|
| port | MAC Address |
| 1 | HGW, L2SW, PLC1, PLC2, A, B, D, E |
| 2 | C |

| PLC1 | |
|---|---|
| port | MAC Address |
| 1 | HGW, L2SW, WB, A, B, C, D |
| 2 | PLC2, E |

| PLC2 | |
|---|---|
| port | MAC Address |
| 1 | E |
| 2 | HGW, L2SW, WB, PLC1, A, B, C, D |

### 4.2.2 トポロジ検出アルゴリズム

SNMP を用いて NW 機器から収集した FDB からリンクレイヤのトポロジを検出する研究が存在する [21][22]．ネットワークのトポロジの検出にあたり，NW 機器同士が一つの NW 機器を介してでも接続されている場合を「接続」，一つの物理線で接続されている場合を「直接接続」されていると定義している．また，トポロジの検出に用いる条件式内の表記法について以下のように定義されている．

- $C_{ip}$：NW 機器 $i$ のポート $p$ に接続されている NW 機器の集合
- $D_{jq}$：NW 機器 $j$ のポート $q$ に直接接続されている NW 機器

直接接続を検出するため，以下の三つの条件式が示されている．

- 条件 1 : $C_{ip} = \{j\} \Rightarrow D_{ip} = j$
- 条件 2 : $(D_{ip} = j) \wedge (C_{jq} \ni i) \Rightarrow D_{jq} = i$
- 条件 3 : $(C_{ip} \ni j) \wedge (C_{jq} \ni i) \wedge (C_{ip} \cap C_{jq}) \neq \phi \Rightarrow D_{ip} \neq j$

ネットワークのトポロジを検出するためには，NW 機器同士が直接接続されているかどうかを決定する必要がある．まず NW 機器間の接続の関係については，表 4.1 から抽出することが可能である．表 4.2 に図 4.1 の NW 機器間の接続関係を示す．

表 4.2: NW 機器間の接続関係

| | | Connection Src | | | | |
|---|---|---|---|---|---|---|
| | | HGW | L2SW | WB | PLC1 | PLC2 |
| Connection Dst | HGW | - | 4 | 1 | 1 | 2 |
| | L2SW | 1 | - | 1 | 1 | 2 |
| | WB | 2 | 4 | - | 1 | 2 |
| | PLC1 | 1 | 3 | 1 | - | 2 |
| | PLC2 | 1 | 3 | 1 | 2 | - |

NW 機器間の直接接続を決定するには，以下の (1)〜(3) を新たな直接接続を決定できなくなるまで繰り返す．

(1) 条件 1 を表 4.2 に適用し，各列の中で唯一に決まるポートを選択し，直接接続を決定する．

(2) (1) の結果について条件 2 を適用する．

(3) (2)(3) の結果から同一ポートに接続される他の NW 機器は直接接続ではない．

図 4.1 のネットワークを上記の条件式を実際に適用していくと図 4.2 のようになる．このトポロジでは，8 ステップで NW 機器のみからなるトポロジの直接接続関係を検出できていることがわかる．図中の丸で囲われた数字は検出された直接接続関係を表し，バツで上書きされている数字は，直接接続関係ではないことを表している．

条件 1： $C_{ip}=\{j\} \Rightarrow D_{ip}=j$
$C_{HGW\,2}=\{WB\} \Rightarrow D_{HGW\,2}=WB$

条件 2：$(D_{ip}=j) \wedge (C_{jq} \ni i) \Rightarrow D_{jq}=i$
$(D_{HGW\,2}=WB) \wedge (C_{WB\,1} \ni HGW) \Rightarrow D_{WB\,1}=HGW$

① Connection Src

| Connection Dst | HGW | L2SW | WB | PLC1 | PLC2 |
|---|---|---|---|---|---|
| HGW | - | 4 | 1 | 1 | 2 |
| L2SW | 1 | - | 1 | 1 | 2 |
| WB | (2) | ~~4~~ | - | ~~1~~ | ~~2~~ |
| PLC1 | 1 | 3 | 1 | - | 2 |
| PLC2 | 1 | 3 | 1 | 2 | - |

② Connection Src

| Connection Dst | HGW | L2SW | WB | PLC1 | PLC2 |
|---|---|---|---|---|---|
| HGW | - | 4 | (1) | 1 | 2 |
| L2SW | 1 | - | ~~1~~ | 1 | 2 |
| WB | (2) | ~~4~~ | - | ~~1~~ | ~~2~~ |
| PLC1 | 1 | 3 | ~~1~~ | - | 2 |
| PLC2 | 1 | 3 | ~~1~~ | 2 | - |

条件 1： $C_{ip}=\{j\} \Rightarrow D_{ip}=j$
$C_{PLC1\,2}=\{PLC2\} \Rightarrow D_{PLC1\,2}=PLC2$

条件 2：$(D_{ip}=j) \wedge (C_{jq} \ni i) \Rightarrow D_{jq}=i$
$(D_{PLC1\,2}=PLC2) \wedge (C_{PLC2\,2} \ni PLC1) \Rightarrow D_{PLC2\,2}=PLC1$

③ Connection Src

| Connection Dst | HGW | L2SW | WB | PLC1 | PLC2 |
|---|---|---|---|---|---|
| HGW | - | 4 | (1) | 1 | 2 |
| L2SW | 1 | - | ~~1~~ | 1 | 2 |
| WB | (2) | ~~4~~ | - | ~~1~~ | ~~2~~ |
| PLC1 | 1 | 3 | ~~1~~ | - | 2 |
| PLC2 | ~~1~~ | ~~3~~ | ~~1~~ | (2) | - |

④ Connection Src

| Connection Dst | HGW | L2SW | WB | PLC1 | PLC2 |
|---|---|---|---|---|---|
| HGW | - | 4 | (1) | 1 | ~~2~~ |
| L2SW | 1 | - | ~~1~~ | 1 | ~~2~~ |
| WB | (2) | ~~4~~ | - | ~~1~~ | ~~2~~ |
| PLC1 | 1 | 3 | ~~1~~ | - | (2) |
| PLC2 | ~~1~~ | ~~3~~ | ~~1~~ | (2) | - |

条件 1： $C_{ip}=\{j\} \Rightarrow D_{ip}=j$
$C_{L2SW\,4}=\{HGW\} \Rightarrow D_{L2SW\,4}=HGW$

条件 2：$(D_{ip}=j) \wedge (C_{jq} \ni i) \Rightarrow D_{jq}=i$
$(D_{L2SW\,4}=HGW) \wedge (C_{HGW\,1} \ni L2SW) \Rightarrow D_{HGW\,1}=L2SW$

⑤ Connection Src

| Connection Dst | HGW | L2SW | WB | PLC1 | PLC2 |
|---|---|---|---|---|---|
| HGW | - | (4) | (1) | ~~1~~ | ~~2~~ |
| L2SW | 1 | - | ~~1~~ | 1 | ~~2~~ |
| WB | (2) | ~~4~~ | - | ~~1~~ | ~~2~~ |
| PLC1 | 1 | 3 | ~~1~~ | - | (2) |
| PLC2 | ~~1~~ | ~~3~~ | ~~1~~ | (2) | - |

⑥ Connection Src

| Connection Dst | HGW | L2SW | WB | PLC1 | PLC2 |
|---|---|---|---|---|---|
| HGW | - | (4) | (1) | ~~1~~ | ~~2~~ |
| L2SW | (1) | - | ~~1~~ | 1 | ~~2~~ |
| WB | (2) | ~~4~~ | - | ~~1~~ | ~~2~~ |
| PLC1 | ~~1~~ | 3 | ~~1~~ | - | (2) |
| PLC2 | ~~1~~ | ~~3~~ | ~~1~~ | (2) | - |

条件 1： $C_{ip}=\{j\} \Rightarrow D_{ip}=j$
$C_{L2SW\,3}=\{PLC1\} \Rightarrow D_{L2SW\,3}=PLC1$

条件 2：$(D_{ip}=j) \wedge (C_{jq} \ni i) \Rightarrow D_{jq}=i$
$(D_{L2SW\,3}=PLC1) \wedge (C_{PLC1\,1} \ni L2SW) \Rightarrow D_{PLC1\,1}=L2SW$

⑦ Connection Src

| Connection Dst | HGW | L2SW | WB | PLC1 | PLC2 |
|---|---|---|---|---|---|
| HGW | - | (4) | (1) | ~~1~~ | ~~2~~ |
| L2SW | (1) | - | ~~1~~ | 1 | ~~2~~ |
| WB | (2) | ~~4~~ | - | ~~1~~ | ~~2~~ |
| PLC1 | ~~1~~ | (3) | ~~1~~ | - | (2) |
| PLC2 | ~~1~~ | ~~3~~ | ~~1~~ | (2) | - |

⑧ Connection Src

| Connection Dst | HGW | L2SW | WB | PLC1 | PLC2 |
|---|---|---|---|---|---|
| HGW | - | (4) | (1) | ~~1~~ | ~~2~~ |
| L2SW | (1) | - | ~~1~~ | (1) | ~~2~~ |
| WB | (2) | ~~4~~ | - | ~~1~~ | ~~2~~ |
| PLC1 | ~~1~~ | (3) | ~~1~~ | - | (2) |
| PLC2 | ~~1~~ | ~~3~~ | ~~1~~ | (2) | - |

図 4.2: トポロジ検出アルゴリズムの適用

この結果から表 4.3 の NW 機器間の直接接続関係を得る．図 4.3 に検出された NW 機器

のみからなるトポロジを示す.

表 4.3: NW 機器間の直接接続関係

| Connection Src | | Connection Dst |
|---|---|---|
| Device Category | Port Number | Device Category |
| HGW | 2 | WB |
| WB | 1 | HGW |
| PLC1 | 2 | PLC2 |
| PLC2 | 2 | PLC1 |
| HGW | 1 | L2SW |
| L2SW | 4 | HGW |
| L2SW | 3 | PLC1 |
| PLC1 | 1 | L2SW |

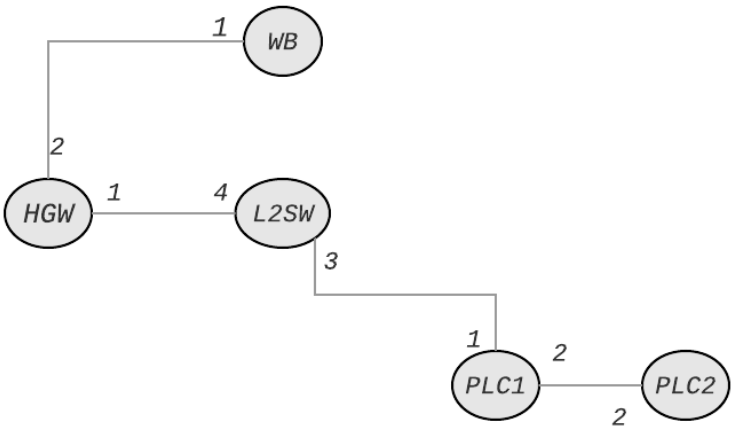


図 4.3: NW 機器のみからなるトポロジ

エンド端末に関しては，図 4.4 のように，各 FDB から NW 機器の接続元ポートの接続先エンド端末が一つのみの場合，直接接続されていることを検出することができる．図中の丸で囲まれた数字が直接接続を検出されたエンド端末を示している．

| HGW | |
|---|---|
| port | MAC Address |
| 1 | L2SW, PLC1, PLC2, B, D, E |
| 2 | WB, C |
| 3 | A |

| L2SW | |
|---|---|
| port | MAC Address |
| 1 | D |
| 2 | B |
| 3 | PLC1, PLC2, E |
| 4 | HGW, WB, A, C |

| WB | |
|---|---|
| port | MAC Address |
| 1 | HGW, L2SW, PLC1, PLC2, A, B, D, E |
| 2 | C |

| PLC1 | |
|---|---|
| port | MAC Address |
| 1 | HGW, L2SW, WB, A, B, C, D |
| 2 | PLC2, E |

| PLC2 | |
|---|---|
| port | MAC Address |
| 1 | E |
| 2 | HGW, L2SW, WB, PLC1, A, B, C, D |

図 4.4: エンド端末に関する処理

条件 1，2 で直接接続を決定できない場合がある．図 4.5 に示す NW 機器のみからなるトポロジについて考える．NW 機器 A から情報を完全に得られない場合，HGW と NW 機器 B とが直接接続していることになり，NW 機器 A の存在がわからなくなるだけである．一方で，NW 機器 B から情報が完全に得られない場合，条件 1 を用いて NW 機器 A，C 間の直接接続を決定するだけで，NW 機器 A，C，D 間の直接接続を決定することはできない．これは，NW 機器間 A，C，D 間で接続媒体を共有しているようなネットワークでも同様のことが言える．そこで，条件 3 を用いることで，条件 1，2 だけでは検出できなかった直接接続を決定できる場合がある．

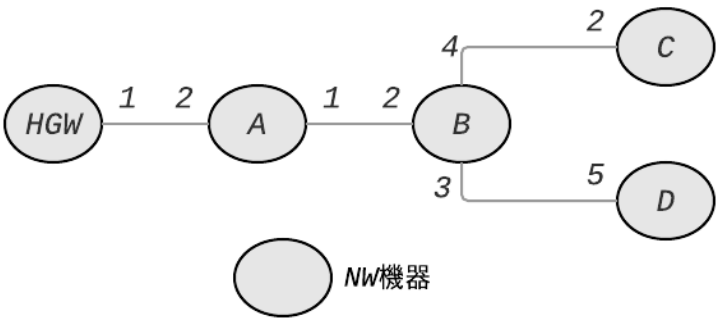


図 4.5: ネットワークトポロジ例

- $(C_{PLC22} \ni PLC3) \wedge (C_{PLC31} \ni PLC2) \wedge (C_{PLC22} \cap C_{PLC31}) \neq \phi \Rightarrow D_{PLC22} \neq PLC3$
- $(C_{PLC12} \ni PLC3) \wedge (C_{PLC31} \ni PLC1) \wedge (C_{PLC12} \cap C_{PLC31}) \neq \phi \Rightarrow D_{PLC12} \neq PLC3$
- $(C_{PLC12} \ni PLC2) \wedge (C_{PLC22} \ni PLC1) \wedge (C_{PLC12} \cap C_{PLC22}) \neq \phi \Rightarrow D_{PLC12} \neq PLC2$

## 4.3 ホームネットワークにおけるネットワーク記述モデル

### 4.3.1 ネットワーク記述モデルの検討

YANG data model for Network Topology と NetJSON の特徴について比較して検討する．NetJSON はシンプルな記述モデルであるが，YANG data model for Network Topology に比べて拡張性に乏しいという面がある．また，YANG data model for Network Topology は YANG というデータモデルに基づいているため，YANG モデルとして定義することで，JSON や XML などの様々な形式への変換が可能になることから NetJSON に比べて汎用性が高いという面もある．そこで，本研究では，YANG data model for Network Topology 及び NetJSON，それぞれのネットワーク記述モデルを基本として，記述モデルを拡張する形でホームネットワークのトポロジ情報が記述可能なモデルを提案する．

# 第5章　システム設計と実装

前章では，ホームネットワークのトポロジに関連する情報を収集する技術は確立されつつあるが，トポロジを検出する手法についての規定はないことから，既存のトポロジ検出アルゴリズムの適用について述べた．また，検出されたネットワークのトポロジ情報を様々なアプリケーションで利用することを考えた場合，形式的に記述されていることが望ましい．

提案手法に基づいて，ホームネットワークにおけるトポロジ情報の記述を行うためには，まず収集されたトポロジ関連の情報からトポロジを検出する．次に，検出されたトポロジ情報のモデル化をする．トポロジ情報のモデル化を行うことで，システムにおけるデータの交換を可能とし，利便性を向上させることができる．以下に提案手法に基づく，トポロジ情報の記述に必要な事項を挙げる．

- 収集されたトポロジ関連の情報からトポロジ検出を行う
- 検出されたトポロジを記述するためにトポロジ情報のモデル化を行う

本章では，これらを踏まえたシステムの設計について述べる．

## 5.1　トポロジ情報の記述に必要な要素

ホームネットワークのトポロジを記述するには，少なくとも Chassis ID と機器情報 (区分，型番，メーカーコード，機種名)，接続構成情報 (ポート番号，ポート種別，ノードの接続関係) が必要である．また，ホームネットワークの特徴としてユーザが自由に接続する機器を変更できることから，構成は絶えず変化しており定期的な情報の収集が必要になる．

## 5.2　システム全体の構成

図 5.1 にシステム全体の構成を示す．図中の管理運用情報統合のデータベース (DB) は，トポロジ情報のみならず障害情報など，HTIP や SNMP などのプロトコルから得られる管理運用情報を統合するためのシステムであり，同研究室で開発が進められている．本研究では，橙色の部分に重点を置いている．なお，以下の番号は図 5.1 中の番号と対応している．

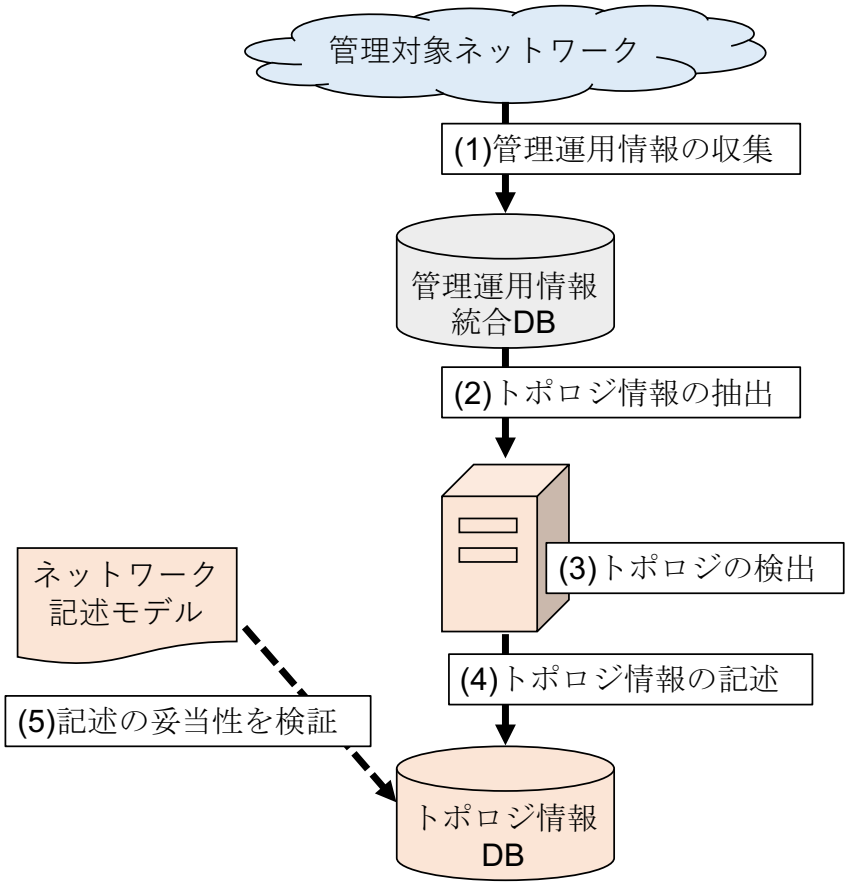


図 5.1: トポロジ検出システムの概要

(1) 管理運用情報の収集

- 管理対象のホームネットワークから HTIP などの管理運用プロトコルを用いて情報を一箇所 (管理運用情報統合 DB) に集約する．

(2) トポロジ情報の抽出

- 管理運用情報統合 DB から管理対象のホームネットワークトポロジを把握するために必要なトポロジ情報を抽出する．

(3) トポロジの検出

- 抽出されたトポロジ情報のうち接続構成情報 (FDB) を使用してトポロジ検出アルゴリズムを実行し，トポロジを検出する．

(4) トポロジ情報の記述

- 検出したトポロジ情報を作成したネットワーク記述モデルに即して記述する．

(5) 記述の妥当性の検証

- ネットワーク記述モデルをスキーマ (例:JSON スキーマ) として定義し，既存のバリデータを用いて検証する．

## 5.3 プロトタイプシステムの実装

トポロジ検出システムのプロトタイプとして，収集されたトポロジ情報からトポロジを検出した後，提案するネットワーク記述モデルに基づいてトポロジ情報を格納し，それらを可視化するアプリケーションを作成した．データ記述言語には JSON を使用した．図 5.2 にトポロジビューアのシステム概要を示す．また，図 5.3 にトポロジビューアの動作イメージを示す．

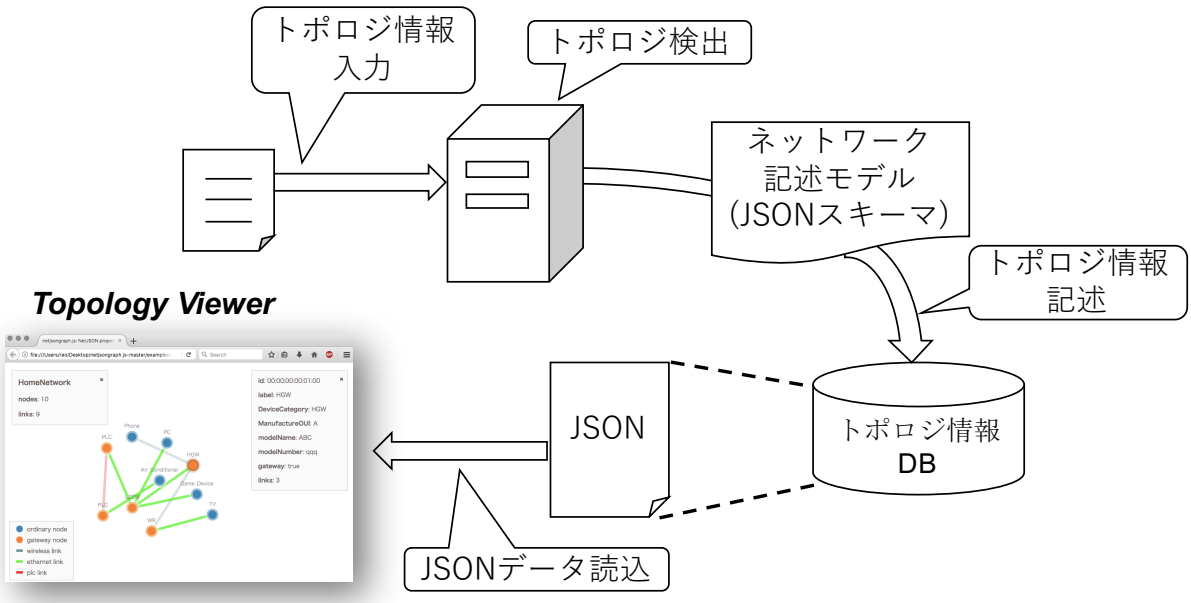


図 5.2: トポロジビューアのシステム概要

ここでは，図 4.1 に示したネットワークのトポロジ情報を入力としている．トポロジを描画する前処理として，トポロジ情報のうち FDB の情報からトポロジ検出アルゴリズムを用いてトポロジを検出している．その後，提案するホームネットワークに対応した記述モデルに基づいた，JSON データを生成している．なお，今回使用したネットワーク記述モデルは NetJSON を拡張したモデルである．最後に，ウェブブラウザ上で動的コンテンツを描画する JavaScript ライブラリ D3.js を用いてトポロジを可視化している．図 5.3 はノードとリンクの種別を色で表示している．また，ノードをクリックすることで，その機器の情報である機器名や型番などを一覧で表示することができる．トポロジ検出システムが保持しているトポロジ情報をいかにユーザに提示するかは，今後の検討課題の一つである．

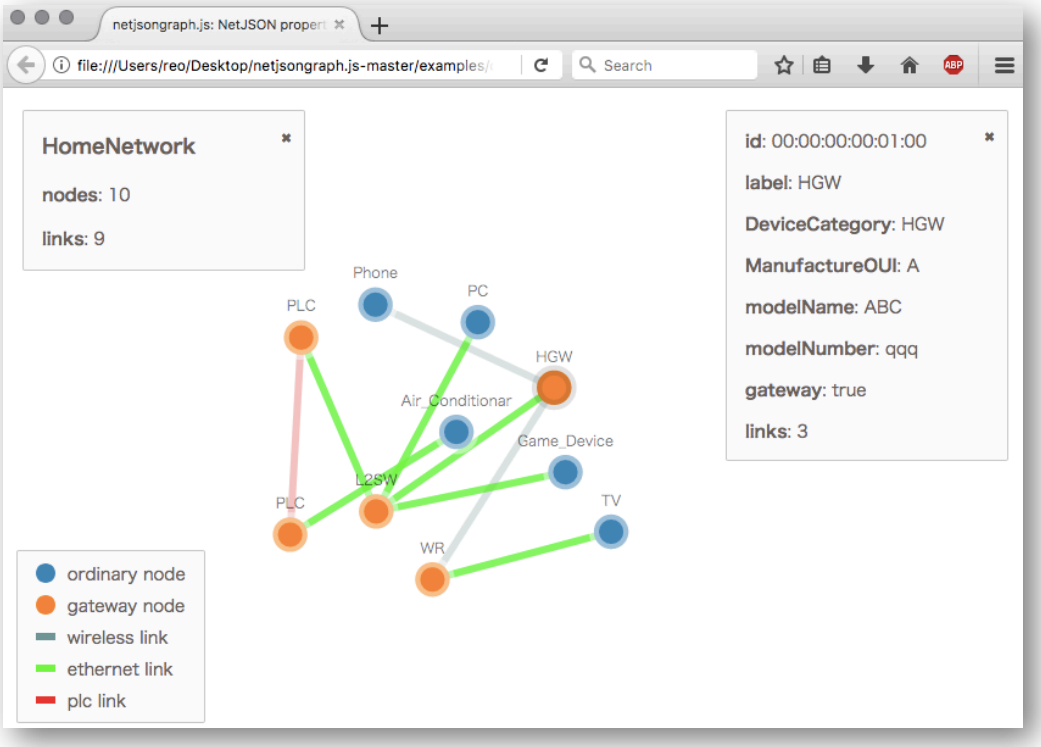


図 5.3: トポロジビューア

# 第6章　評価と考察

前章では，提案手法に基づいたシステムの実装について述べた．本章では，その提案手法に基づいて実装したシステムについて，提案手法の適用性，性能評価の面から評価をする．

## 6.1　提案システムの適用性

提案手法は，ホームネットワークのネットワーク構成を正確に把握することを目的としている．提案手法では，ネットワークの設定情報やネットワーク資源情報を記述する方法については提供していない．しかし，ホームネットワークを管理し運用をしていく上で必要になるのは，ネットワークのトポロジ情報だけではなく，ネットワークの設定に関する情報も重要になる．また，ネットワークの不具合発生時には，ネットワーク回線の帯域や機器の処理性能等のネットワーク資源に関わる情報についても把握できることが望まれる．

提案手法では，近年増えつつある IP 技術を使用しない Zigbee や Bluetooth といったワイヤレスセンサネットワークには対応することができない．

## 6.2　性能評価

ホームネットワークのトポロジ情報の記述を行う前に，接続構成情報からトポロジの検出をする．トポロジの検出に要する時間は可能な限り短い方が良い．そこで，提案手法にあるトポロジ検出アルゴリズムの計算時間を測定する．

まず実験を行うにあたり，実験用のネットワークを作成する．ネットワーク機器のみからなるトポロジの検出をするための，検証用のネットワークとして，最初に頂点数 $n$ とルートノード $r$ を構築する．そして，ランダム数 $c \in \{1, 2, 3, 4, 5\}$ を求める．そして，ルートノード $r$ に対し，$c$ 個の子ノードを連結する．これ以降は，頂点数が $n$ になるまで，次

の処理を繰り返していく．現在の木に対してランダムに葉ノード $v$ を選択し，ランダム数 $c \in \{1, 2, 3, 4, 5\}$ を決定する．そして，$v$ に $c$ 個の子ノードを追加する．

前章のプロトタイプシステムで実装した，トポロジ検出システムを利用する．実験を行う計算機の構成を以下に示す．

- OS: macOS Sierra 10.12.6
- CPU: 2.2 GHz Intel Core i7
- Memory: 8 GB 1600 MHz DDR3

図 6.1 にトポロジ検出アルゴリズムの計算時間の結果をグラフで示す．

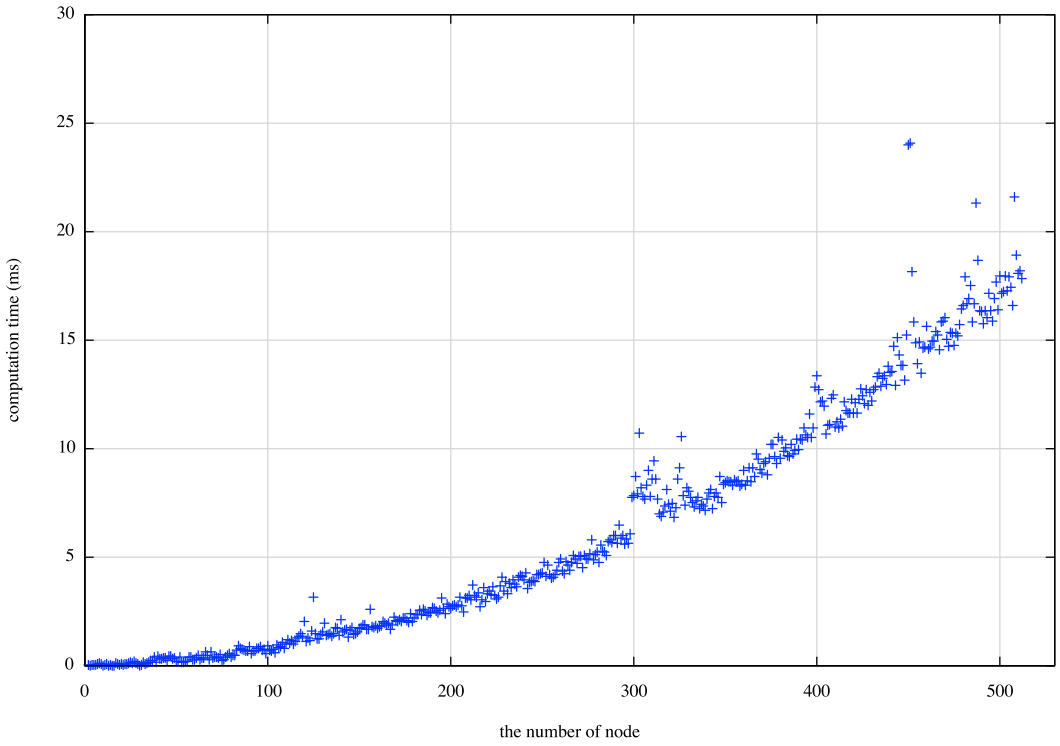


図 6.1: トポロジ検出アルゴリズムの計算時間

# 第7章　おわりに

本章では，本論文におけるまとめと今後の課題について述べる．

## 7.1　まとめ

ホームネットワークでは，異なるネットワークや多種多様な機器が接続されることで，ホームネットワークのトポロジが複雑化している．これに対し，ホームネットワークに接続された各機器より収集された接続構成情報からトポロジを自動的に検出し，ネットワーク記述モデルによってトポロジ情報を記述する手法を提案した．これにより，多岐にわたる端末間，或いはネットワーク技術間のトポロジをレイヤ横断的に記述することができ，ホームネットワークのトポロジ全体を把握することを可能とする．また，ネットワーク記述モデルを利用することにより，トポロジ情報を計算機で読込み可能な形式となる．これにより，ホームネットワークの障害原因の特定を容易化・自動化することが可能になり，ホームネットワークの可用性の向上に貢献できる．

## 7.2　今後の課題

今後の課題として，トポロジを自動的に検出し，形式的に記述して可視化する手法をプロトタイプシステムとして実装した．しかし，システムで利用している接続構成情報はあらかじめ用意した情報であり，HTIP が実装された実際のホームネットワーク環境，或いはシミュレーションされた環境において実験を行う必要があると考えている．

# 謝辞

本研究を行うにあたり，多大なるご指導とご厚状を賜りました丹 康雄教授に深く感謝致します．

また審査をお引き受け頂いた本学 篠田 陽一教授，本学 リム 勇仁准教授，Razvan Beuran 特任准教授には，本論文を執筆するにあたり多大なご助言を頂きました．深く感謝いたします．

副テーマにおいてご指導ご鞭撻を賜りました本学 金子 峰雄教授，本学 大西 正輝客員准教授に深く感謝致します．

本論文をまとめるにあたりご協力を頂いた丹研究室，リム研究室の皆様に厚く御礼申し上げます．

最後に，私の研究に対し理解を示して頂き，学生生活を送るにあたって経済的，精神的に支援して下さった両親に心より感謝を致します．

# 参考文献


[1] J. D. Case，M. Fedor，M. L. Schoffstall and J. Davin: Simple network management protocol (SNMP)，RFC 1157, IETF，1990.

[2] R. Presuhn，J. Case，K. McCloghrie，M. Rose and S. Waldbusser: Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)，RFC3418，IETF，2002.

[3] M. Rose and K. McCloghrie: Structure and Identification of Management Information for TCP/IP-based Internets，RFC1155，IETF，1990.

[4] Changing from ASN.1:1988 to ASN.1:2002，ITU-T. http://www.itu.int/ITU-T/studygroups/com17/changing-ASN/

[5] Link Layer Topology Discovery，Microsoft, 2010. https://msdn.microsoft.com/ja-jp/library/gg156955(v=winembedded.70).aspx

[6] IEEE Computer Society: 802.1AB-2009: Local and Metropolitan Area Networks - Station and Media Access Control Connectivity Discovery，2009.

[7] UPnP Device Architecture1.1，UPnP Forum，2008. http://upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v1.1.pdf

[8] JJ-300.00 ホーム NW 接続構成特定プロトコル第 3.0 版，2017. http://www.ttc.or.jp/jp/document_list/pdf/j/STD/JJ-300.00v3.pdf

[9] A. Bierman and K. Jones: Physical Topology MIB，RFC2922，IETF，2000.

[10] J.J. van der Ham，F. Dikstra，F. Travostino，H. Andree and C. de Laat: Using RDF to Describe Networks，Future Generation Computer Systems，22(8):862-867，2006.

[11] Network Markup Language，NML Working Group，2013. https://www.ogf.org/documents/GFD.206.pdf

[12] G.P. Koslovski，P.V.B. Primet and A.S. Charão: VXDL: Virtual resources and interconnection networks description language，Networks for Grid Application，pp. 138-154，2009.

[13] M. Ghijsen，J. Van der Ham，P. Grosso and C. De Laat: Towards an InfrastructureDescription Language for Modeling Computing Infrastructures, Proc. of the 10th IEEE International Symposium on Parallel and Distributed Processing with Applicationn，pp. 207-214，2012.

[14] NOVI (Network Innovation Over Virtualized Infrastructures). http://www.fp7-novi.eu.

[15] GEYSERS. https://www.geysers.eu.

[16] A.Clemm，J.Medved，R.Varga，N.Bachadur，H.Ananthakrishnan and X.Liu: A Data Model for Network Topologies，Work in Progress，IETF，2017. https://www.ietf.org/id/draft-ietf-i2rs-yang-network-topo-20

[17] M. Bjorklund: YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)，RFC6020，IETF，2010.

[18] M.Bjorklund: The YANG 1.1 Data Modeling Language，RFC7950，IETF，2016.

[19] NetJSON: data interchange format for networks，2015. http://netjson.org/rfc.html

[20] T.Bray: The JavaScript Object Notation (JSON) Data Interchange Format，RFC8259，IETF，2017.

[21] 吉田和幸，ネットワークトポロジの発見とその表示について，大学情報システム環境研究，VOL.10，pp68-74，2007.

[22] 河野優，釜崎正吾，平川龍，大浦昇，吉田和幸，Layer2 ネットワ ーク構成情報の検出アルゴリズムの改良について，情報処理学会「分散システム/インターネット運用技術シンポジウム 2005」論文集 pp. 61-66，2005.