



2008 年度上期未踏 IT 人材発掘・育成事業(未踏ユース)採択案件評価書

1. 担当PM

竹内 郁雄 PM(東京大学大学院 情報理工学系研究科 創造情報学専攻 教授)

2. 採択者氏名

チーフクリエイター: 水谷 后宏(奈良先端科学技術大学院大学 情報科学研究科
情報システム学専攻博士前期課程 1 年 インターネットアーキテクチャ講座)
コクリエイター: なし

3. プロジェクト管理組織

株式会社 オープンテクノロジーズ

4. 委託金支払額

2,607,119 円

5. テーマ名

Programmable-OverlayNetwork の開発

6. 関連Webサイト

なし

7. テーマ概要

本提案では、Programmable な OverlayNetwork の構築を可能とするミドルウェアを開発する。ここでの Programmable とは「設計可能」という位置付けで、Programmable-message と呼ぶ制御メッセージの伝搬・解釈により、OverlayNetwork

上の各ノード間のトポロジー・ルーティング、コンテンツの分散などを開発・管理・運用することが可能となる。この Programmable-OverlayNetwork は以下の2つの機能を持つ。

1.全体化

Programmable-message を OverlayNetwork 上の各ノードに伝搬させることにより、各ノードでは Programmable-message の内容に従った振る舞いを行う。これにより、OverlayNetwork 全体のルーティングアルゴリズム・トポロジー・コンテンツ分散の方法を、OverlayNetwork の動作を停止させることなく、いつでも変更可能となる。

2.局所化

Programmable-message を、目的のコンテンツ・ノードに対して、リクエストする際に付加することにより、目的のコンテンツ・ノードへのアクセスに、局所的なルーティングアルゴリズム・トポロジーを適用することができる。これによって、OverlayNetwork の全体的なルーティングアルゴリズム・トポロジーの変更時の予期せぬ動作を、局所的に検証することが可能となる(テストリスクの軽減)。

以上の要件を満たす本提案では、「OverlayNetwork」の開発支援だけでなく、分散アプリケーション(例 P2P アプリケーション)の開発・管理・運用も可能になる。

8. 採択理由

現在、アプリケーションごとに Chord など、いくつかのオーバーレイネットワークが独立して存在しているが、アプリケーションを止めることなく、いつでも変更できるようなオーバーレイネットワークにするために、第7層でのいわば仮想ルータをメッセージ伝播によって必要なホストにインストールしていくという考え方は興味深い。でも、じゃあ、実際このシステムをどういふところで使うんだという問いに、より多くの人を納得させる答えも用意してほしい。

水谷君はいかにもプログラミング能力が高そうな面構えだ。実際、とある実験で Skype もどきのシステムを2週間で作り上げてしまったという。物理学の統一原理に感銘を受け、オーバーレイネットワークでもそのような統一原理を目指したいという彼の主張には、なんと無理矢理な、とちよつとびっくりしたが、そういう大掴みな物の見方を裏付けるしっかりした基礎力と精神的体力(?)があると見た。それに水谷君自身、なによりも元気がいい。GPL で出したいと頑張っている。

これは19年度I期にスーパークリエイターになった西川賀樹君のシステム(分散シス

テムの開発を支援するテストベッド)と関連が深い。西川君のはシミュレータだが、水谷君のは実機ネットワーク上で動作するシステムだ。2人はぜひ仲良くなってほしい。

9. 開発目標

現在、Overlay Network を利用した分散アプリケーションの開発が盛んに行われている。Overlay Network とは、Client-Server Model と異なり、Server のみがサービスの提供を行うのではなく、Client が Server の機能を持ち、Client 間でサービスの提供を行うための機構である。Overlay Network には Unstructured と呼ばれる無作為にノードがコネクションを張るものと、ノードの個体情報(ID)に基づき、規則的にノードがコネクションを張る Structured と呼ばれる形態が存在している。図 1 に Client-Server Model と Overlay Network の概要図を記す。

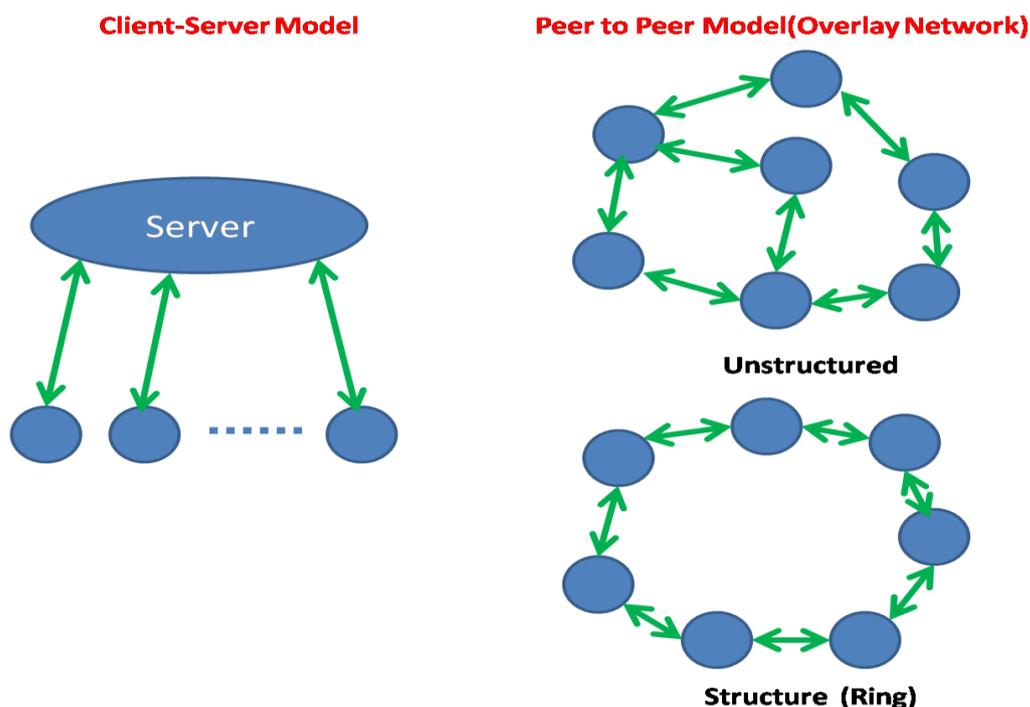


図 1. Client-Server Model と Overlay Network の概要図

Overlay Network を利用したアプリケーションでは、ルーティングなどにバグがあった場合、アプリケーションの動作を止めた上で Overlay Network を再構築する必要があるため、分散アプリケーションの開発・管理・運用に手間がかかるという問題が存在する。

本プロジェクトでは、Overlay Network を用いたアプリケーションの開発、管理、運用を容易に可能とするミドルウェアを作成することを目的とした。

10. 進捗概要

実施計画書では Overlay Network の全体的な変化・局所的な変化を目標にしていた。しかし、本ミドルウェアではこの機能以外にも多数の機能を適用する事ができた。

1. Visualizer の開発

Overlay Network で行っている「ネットワークの仮想化」を直感的に分かりやすく表現、確認するために作成した。加えて Visualizer を入カインタフェースとして拡張し、Visualizer から Overlay Network の全体的な変化・局所的な変化を引き起こす事ができるようになり、動作検証を容易に行う事ができるようになった。また、テキスト形式でノード間の通信ログ、Overlay Network 内で発生した総パケット量(byte)、メッセージ別統計などの情報も得ることができる。Visualizer のスナップショットを図 2 に記す。

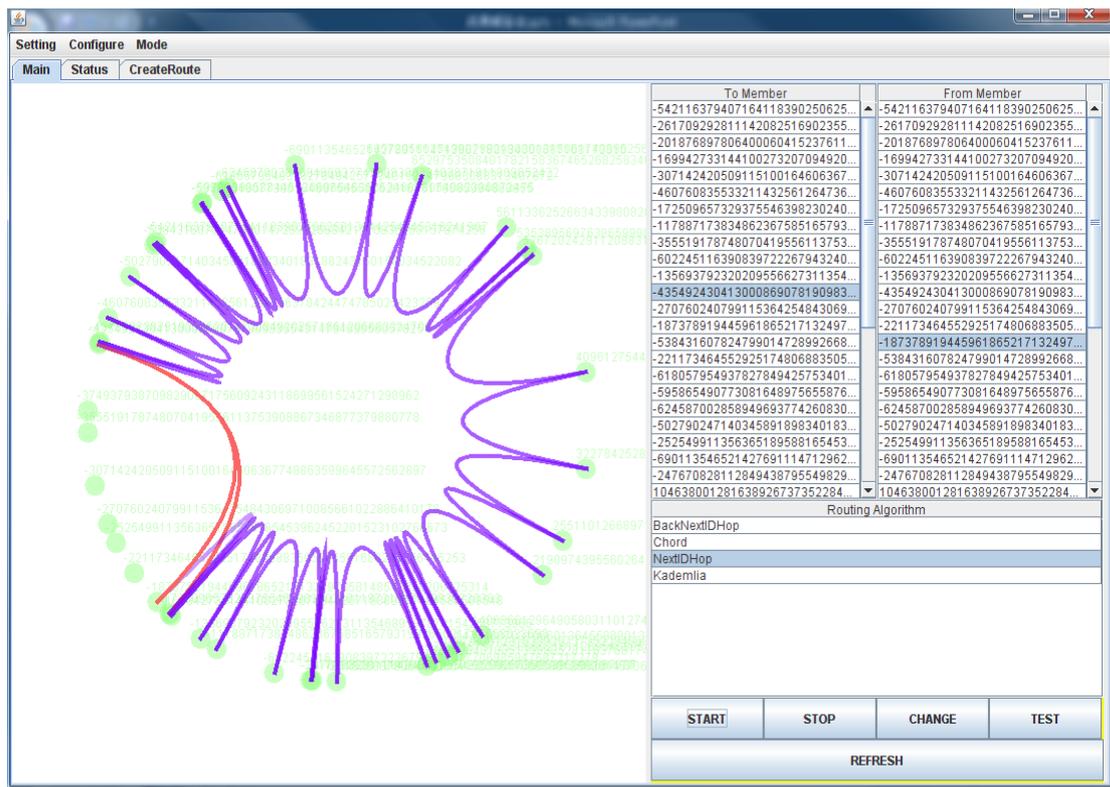


図 2. Visualizer のスナップショット

2. 複数 Overlay Network の並列動作

一つのアプリケーションにおいて Overlay Network の変更を行うだけでなく、複数の Overlay Network を並列して動作させる事が可能となった。これにより、一つ

のアプリケーションのためのミドルウェアだけでなく、検証基盤としてのミドルウェアとなった。本ミドルウェアで構築した検証基盤では、様々な Overlay Network を、どこからでも動作させる事が可能となった。加えて、容易に Overlay Network の構築が行えるように、Overlay Network の API を用意した。

3. Planet-lab を利用した実機実験

従来、提案されてきた Overlay Network の検証機構の大半は、閉じたクラスタ(サーバ群)やエミュレーションでしか有効性を検証していない。しかし、本ミドルウェアはエミュレーション実験だけではなく Planet-lab エラー! 参照元が見つかりません。と呼ばれる世界の学術機関を結んだ実験ネットワークを用いて検証実験を行う事ができた。Planet-lab を利用するに当たり、Planet-lab 上への本ミドルウェアの配布・実行・停止を行うためのスクリプトを作成した。結果的に 100 台近い実機(99 台)を利用した実験を行う事ができた。

11. 成果

従来、提案されてきた Overlay Weaver や peeremu などの Overlay Network の検証機構では、単一の Overlay Network しか一度に検証する事ができず、検証のたびに Overlay Network を再構築する必要があり、検証するためのコストが大きかった。このコストは、実環境上で顕著にあらわれ、世界中に広域に広がるノードに対して再構築を行うには、オペレーション作業が作業効率上極めて大きなボトルネックになってくるので、Overlay Network の検証の敷居が非常に高くなる。本ミドルウェアを用いて検証基盤を構築した場合、Overlay Network の追加や変更手順が抽象化されているため、APIを用いてルーティングアルゴリズムを記述するだけで良い。そのため、実環境であらかじめ検証基盤を構築しておけば、世界中の Overlay Network の研究者、システム開発者が簡単に自身の提案する Overlay Network を検証する事が可能となる。

このような共通検証基盤は OpenDHT と呼ばれるサービスとして、現在 Planet-lab エラー! 参照元が見つかりません。上で稼働している。OpenDHT はルーティングアルゴリズムの検証に利用するのではなく、分散アプリケーションの検証に利用されるため、ルーティングアルゴリズムは完全に固定されている。よってルーティングアルゴリズムレベルでの検証を行う事はできない。本ミドルウェアでは、ルーティングアルゴリズムレベルの検証ができる点でも、分散アプリケーションの検証基盤として優れている。

12. プロジェクト評価

水谷君は根っからのシステム屋である。P2P や Grid などの分散アプリケーションはいろいろなものがあるのだろうが、それらの具体例には目もくれず、ネットワークシステムの土台づくりに一直線である。未踏のコンテキストでは、最終アプリの段階でなにが嬉しくなるかについて少し語ってくれてもいいと思うのだが、それは縁の下の力持ちにとっては雑念とばかりの縁の下への集中だった。オーディションで物理学の統一原理を引合に出して、オーバーレイネットワークの統一原理を語っていたのはやっぱりそうだったのだ。インターネット業界ではその名を知らない人のいない砂原秀樹さんのお弟子さんだが、そんなわけで結構丁々発矢とやりあっているらしい。元気のいいことだ。

提案段階と成果報告段階で、概念そのものは変わらないが用語はかなり変わった。水谷君の英語の使い方がちょっと妙だったので、用語については竹内のほうからかなりのアドバイスをした。多少はわかりやすくなったと思う。

アプリケーション毎にいろいろなオーバーレイネットワークがあり、アプリケーションとオーバーレイネットワークがセットになってしまっている現状は、オーバーレイネットワークの抽象化が不十分ということである。これらを異なる層のものとして分離することができれば、各種の分散アプリケーションの開発・保守・維持の各種コストが低減されるはずだというのが水谷君の基本発想であろう。誰もいままでやってなかったのかしら？というぐらい至極真っ当な発想である（あとで知ったのだが、ほかにもいろいろ研究があるようだ）。

水谷君は、それらを分離するばかりではなく、オーバーレイネットワークを動的に変更したり、オーバーレイネットワーク自体を多数並存させる仕組みを考案し、実現した。

ネットワークポロジの抽象階層の基底にインフラオーバーレイネットワークを置き、その上を流れる RAT メッセージで自由に好きなオーバーレイネットワークをつくれるようにするわけである。このとき、ちょい試しにやってみるという「局所化」という仕掛けも用意して、開発や実験が手軽にできるようにもした。

シミュレータ上でも、実機ネットワーク上でも同じ枠組で動くのが特色だが、最初に実機ネットワークで実験をしようとして大変なことになったらしい（警告メールの嵐）。しかし単純なバグということで PM レビューが終わった直後に簡単に一件落着。砂原さんによれば、これはネットワーク研究者が通り抜けないといけない一種の試練らしいが、とりあえず試練を乗り越えた。ついでに成果も盤石になり、めでたいことである。

さて、この成果が少なくとも研究者たちに広く使われるようになるかどうか、竹内にはまだよくわからない。水谷君もせっかくなつくつたのだから、今度はそちらの方向での努力（ドキュメント化など）をしてほしい。

総じて、やるべきことをきちんと成し遂げたと思う。機能・性能に関しての評価をしっかりやってほしい。ここで開発した技術は将来のスケーラブルなサービス基盤の構築には確実に役立つと思われる。

13. 今後の課題

現在、ノード ID はハッシュ値で表現されている。ハッシュを用いる利点として負荷分散が挙げられる。しかし、ハッシュを用いた ID を用いることによって、IP ネットワーク上では近隣に位置していても、Overlay Network 上では全く異なる Location を持つことになる。この問題を解決するために、ハッシュによるノード ID 以外にも、IP ネットワークの情報(帯域、遅延)を用いて、InfraOverlay Network を構築する事が出来れば、効率的なノード管理ができると考えられる。

本ミドルウェアを検証基盤として用いることで、様々な Overlay Network を構築する事が可能となる。つまり、現在インターネット上に存在しているサービス形態(クライアントサーバシステム、ストリーミングサービス、P2P サービス)を再現し、それらのサービスを共存させた上でサービスごとに動作の検証を行う事が可能となる。これは、従来不可能だった「Overlay Network の並列検証」を可能とした恩恵である。そのため、この検証基盤を用いることで、Overlay Network の設計を、様々なサービスが干渉し合うインターネットをエミュレートする事によりサービス間の干渉を考慮しながら、大局的にできると考えられる。これは NICT/大阪大のチームが行っている PIAX プロジェクトと方向性が一致している部分があり、研究競争・共同研究 などの案が浮上している。