

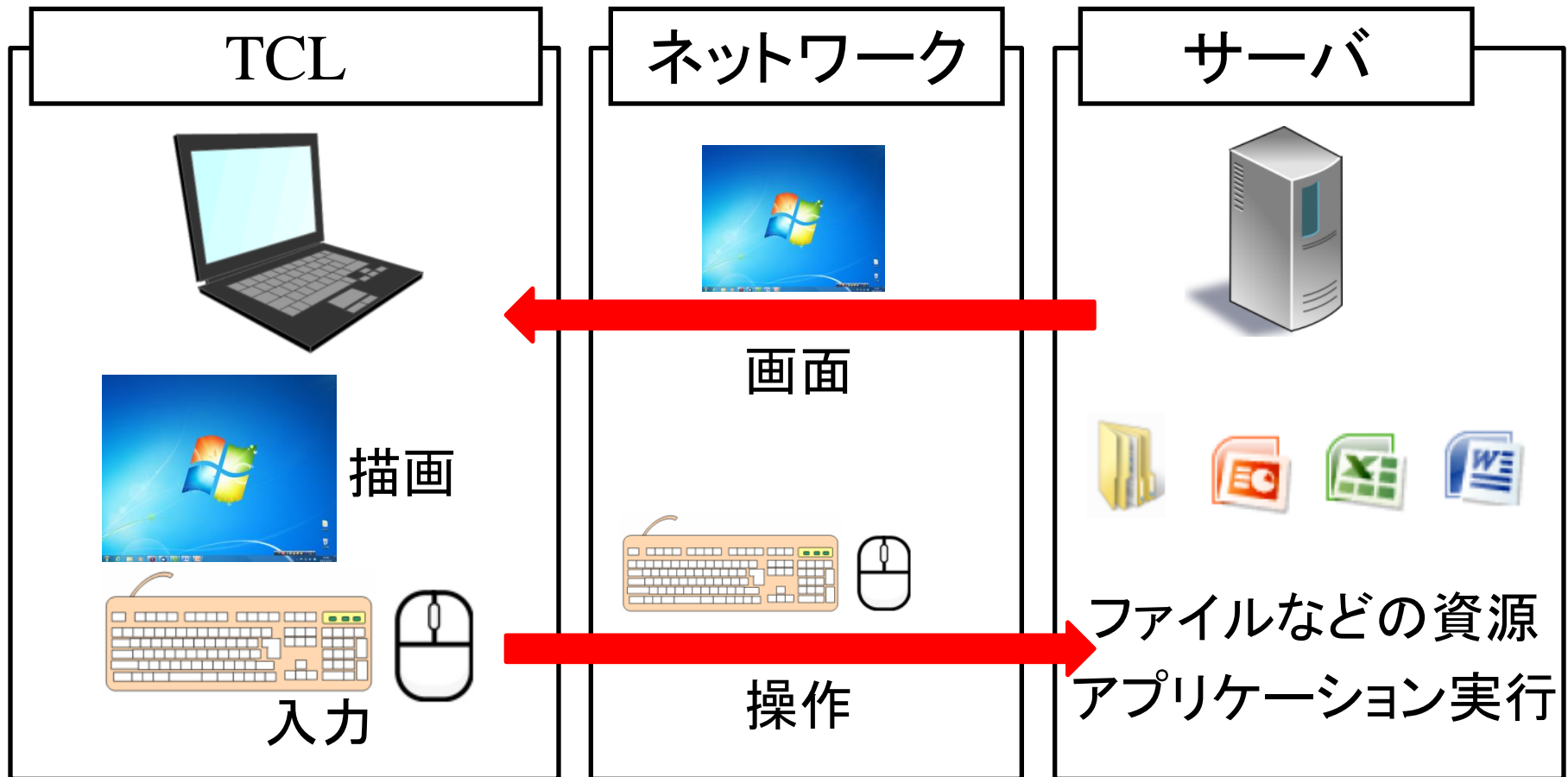
シンクライアントのための 通信状況に応じた仮想マシンの 最適配置手法

53期 情報工学科 研究科生
深堀 秀治

- 仮想マシン型のシンククライアント (TCL) システム
 - クライアントの機能を入出力に絞る, サーバで処理を行う
 - 問題点** 入出力がネットワークを介するため, 性能が低下する
 - 提案** TCLと仮想マシンを近隣に配置
- ライブマイグレーション (LM)
 - 仮想マシンを停止させることなく, 別の物理マシンに移動する
 - 問題点** LAN間でのLM時にネットワーク設定変更が必要
 - 提案** OpenFlowを用いてネットワーク設定変更を自動化

LMとOpenFlowを用いた
仮想マシンの配置システムOVCTを提案, 実装

- クライアントに入出力機能しか持たせず，サーバ側で実際の処理や資源の管理を行うシステム



TCLのユースケース

4/18

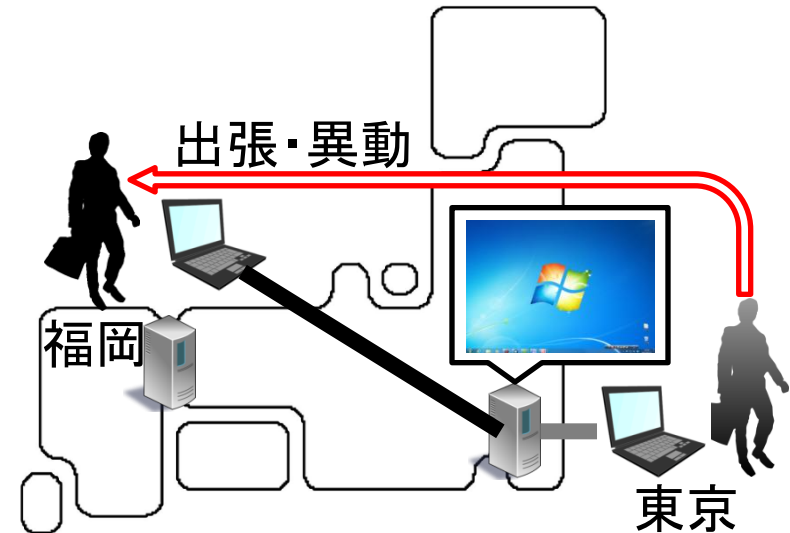
■ ユースケース

■ 通勤などの移動中の利用

- TCLに情報を持たないため、紛失しても情報漏洩に繋がらない

■ 他の拠点への出張・異動

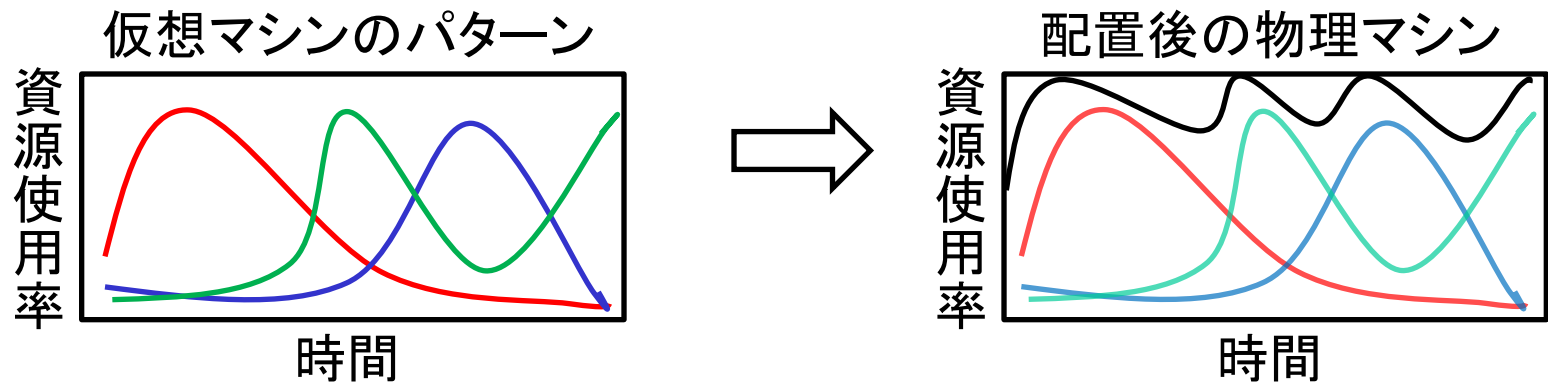
- TCLを持ち出さなくても、出張先に端末があれば、自身の環境を再現可能である



問題点

TCLと仮想マシンの距離に応じて、回線の混雑や減衰による遅延が発生しやすくなり、性能が低下

- TCLシステムにおける仮想マシンの配置
 - 仮想マシンの資源使用量が一定周期でパターン化することを利用して, 各使用パターンの相関を考慮し, 相関の低い仮想マシン同士を同じ物理サーバへ配置することで, 資源の競合を抑える



仮想マシンのパフォーマンスを維持しつつ,
物理マシンの台数を削減することを目的としている

提案: OVCT [Optimal placement method of Virtual machine in accordance with Communication situation for Thin client] 6/18

- TCLと仮想マシンを通信を検知
 - OpenFlowを用いて, ネットワーク全体を監視しTCL一仮想マシン間の接続を検知する

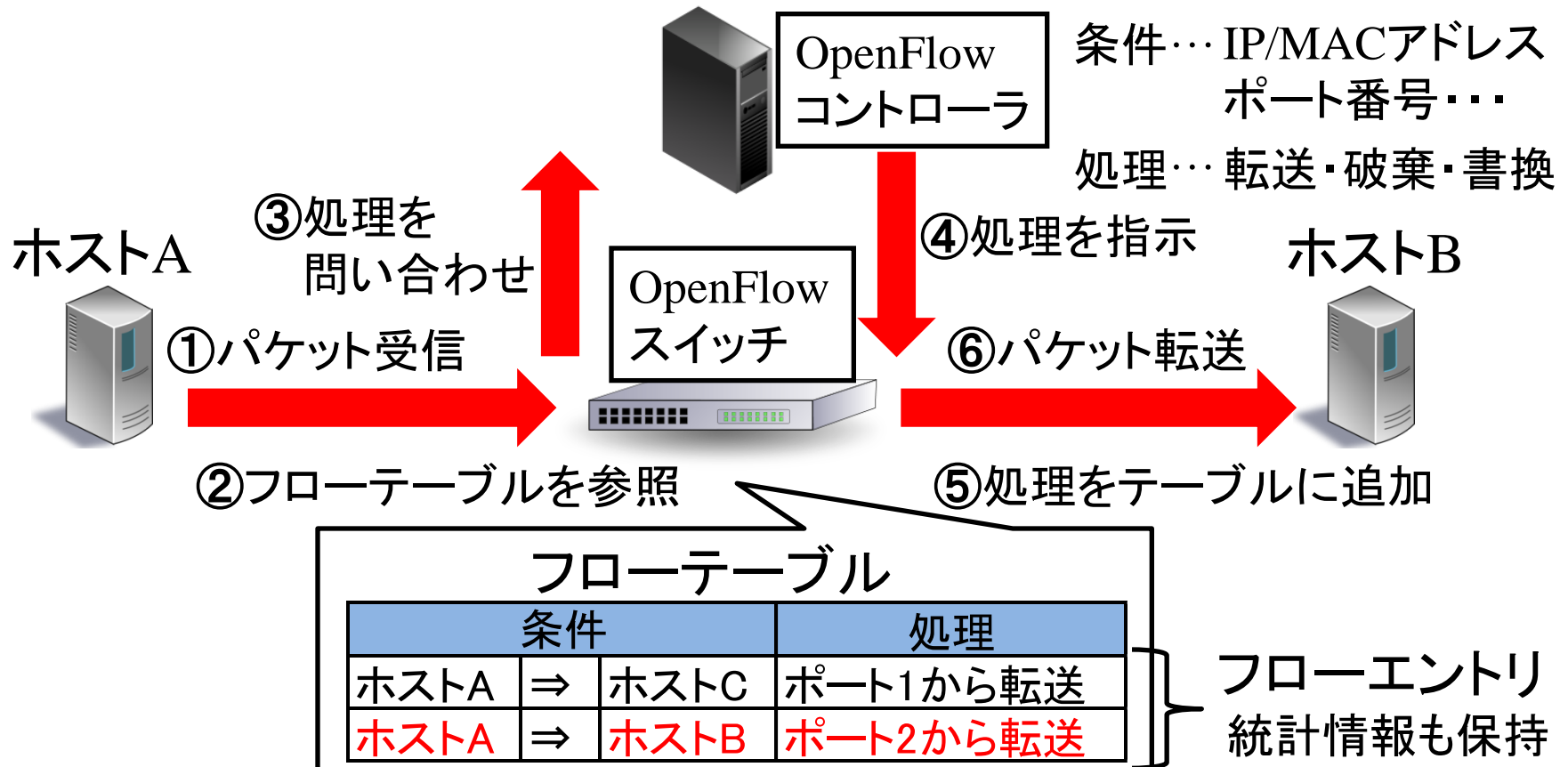
- TCLと仮想マシンを近隣に配置
 - LMを用いて, TCL一仮想マシン間の経路を短縮し性能を向上させる

- LAN間でのLM
 - OpenFlowを用いて, 異なるLANへのLMによって必要になるネットワーク設定の変更を自動化

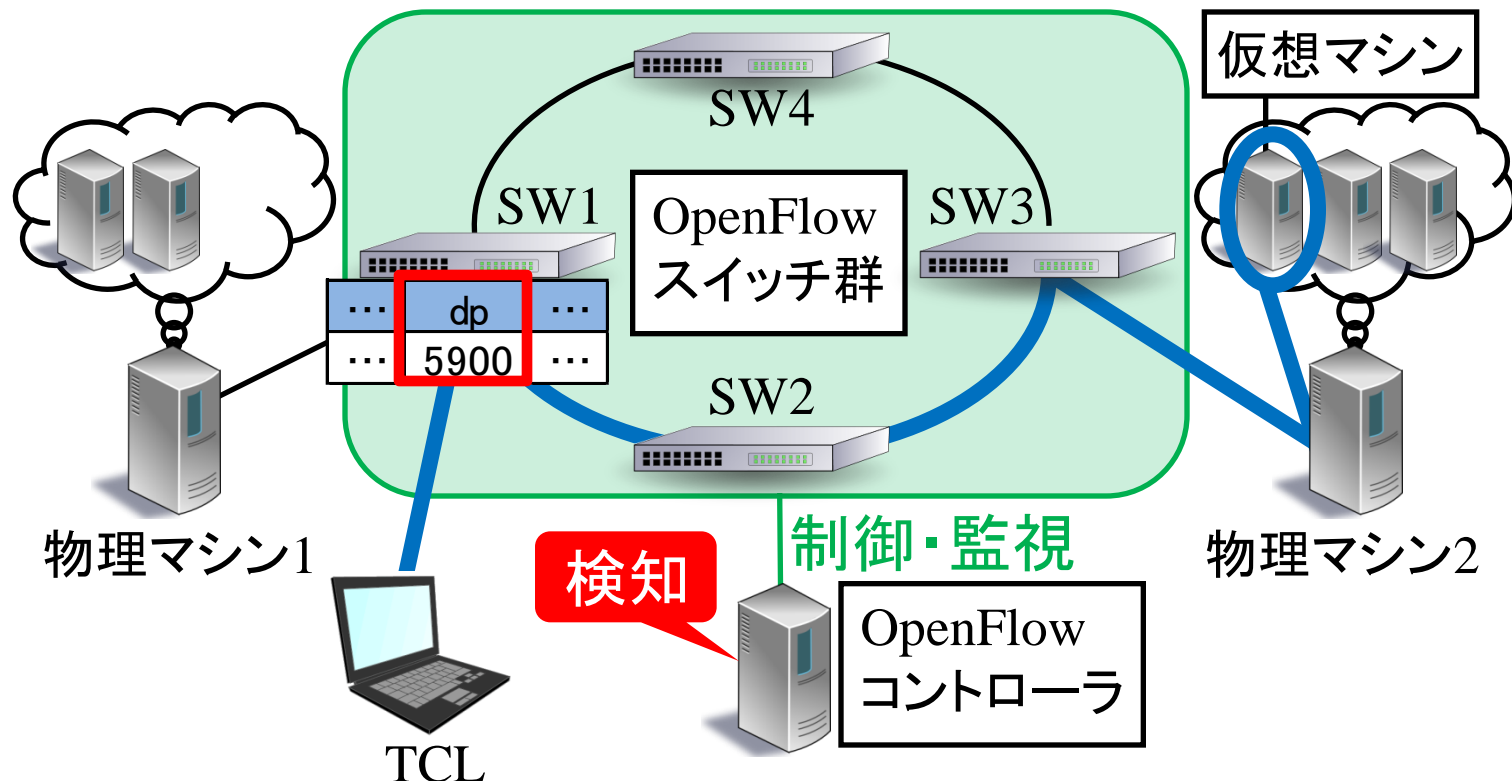
OpenFlow

7/18

- Software Defined Networkの一種
 - ネットワークをソフトウェアで制御する技術
 - 制御部(コントローラ)と駆動部(スイッチ)が独立



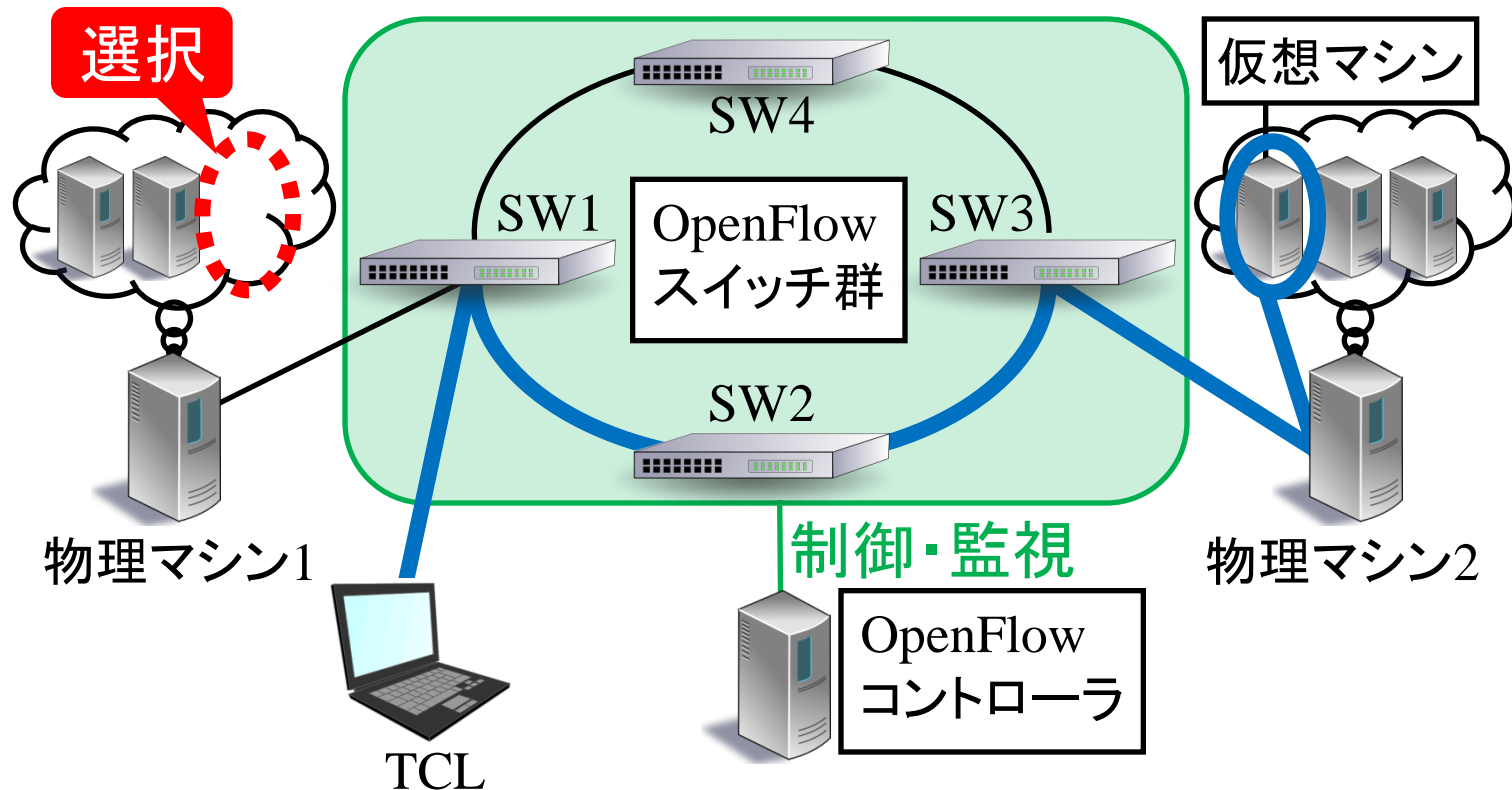
- OpenFlowコントローラでパケットを監視
 - TCLからの宛先ポート番号 $dp = 5900$ (VNC) である通信を検知
 - TCLと仮想マシンが異なるOpenFlowスイッチ配下であった場合、配置場所を検討



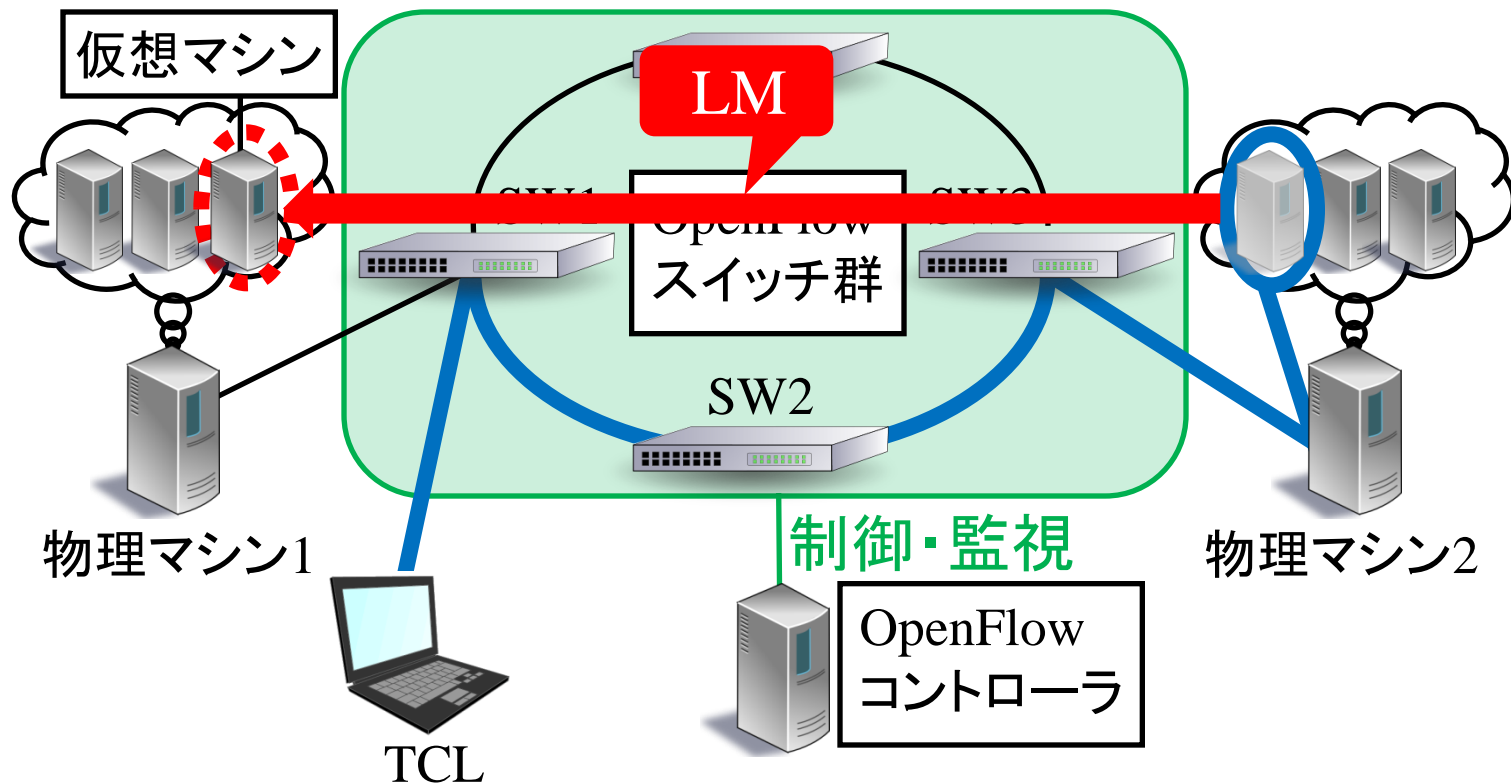
配置場所の検討

9/18

- TCLから経路が最短な物理マシンを選択
 - ネットワーク構成から，経路が最短になる物理マシンを選択
 - 物理マシンのリソースを調べ，空きリソースがあれば選択



- 選択された物理マシンへLM
 - TCL－仮想マシン間の経路を短縮するために仮想マシンをLM

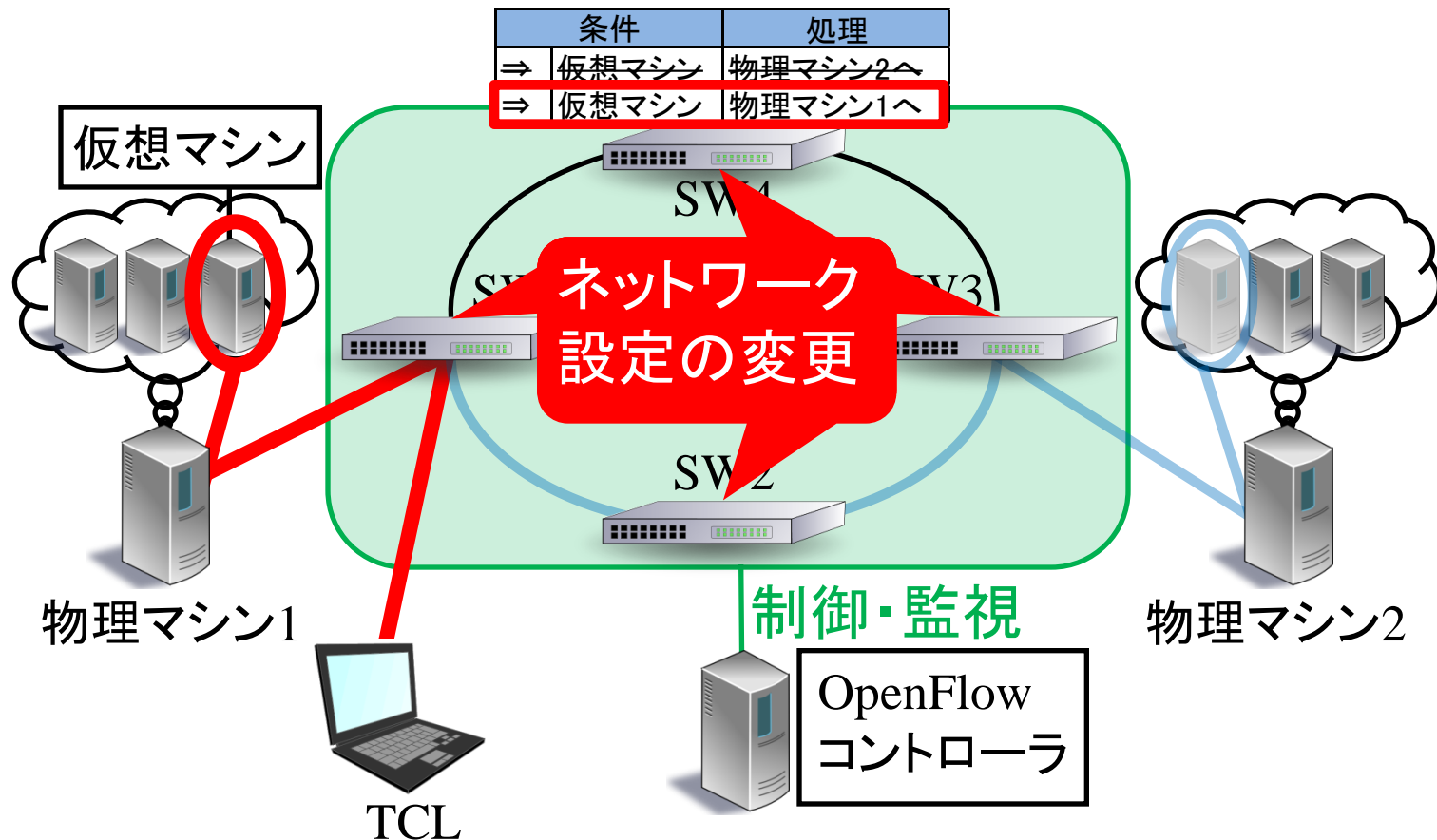


ネットワーク設定の変更

11/18

■ 自動でネットワーク設定の変更

- フローエントリを書換え、仮想マシン宛のパケットの転送先を物理マシン1から物理マシン2に変更



- 提案手法の有用性を確認するため、提案手法を実装

開発環境		数量
OpenFlowコントローラ (OFC)		1台
OS	Ubuntu 12.04	
フレームワーク	trema 0.4.5	
言語	ruby 1.8.7	
プロトコル	OpenFlow 1.0	
OpenFlowスイッチ (OFS)		3台
スイッチ	WHR-G301N	
ファームウェア	OpenFlow 1.0 for WHR-G301N[3]	
マシン		2台
ホストOS	Ubuntu 12.04	
ゲストOS	CentOS 5.3	
ハイパーバイザー	qemu-kvm 1.0	



WHR-G301N[2]



[2] WHR-G301N (<http://buffalo.jp/products/catalog/network/whr-g301n/>)

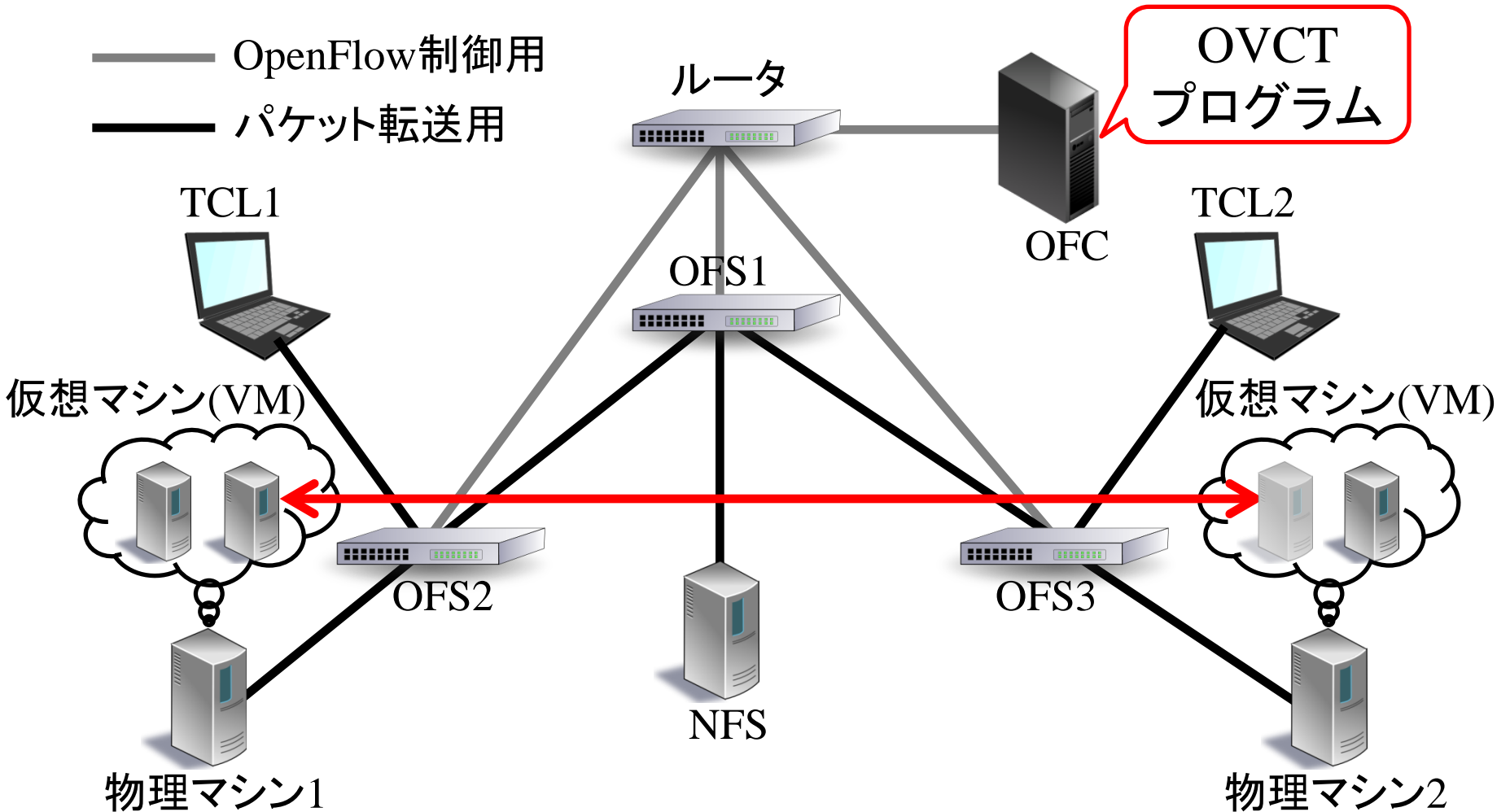
(2014 年2 月現在)

[3] OpenFlow in theBox (<http://openflow.inthebox.info/>)

(2014 年2 月現在)

システム構成

13/18



- 条件

- 前頁のシステム概要・構成と同等

- 評価項目

- TCL一仮想マシン間の通信速度

- TCLの性能

- ネットワーク全体のパケット中継数・量

- ネットワーク負荷

- 比較対象

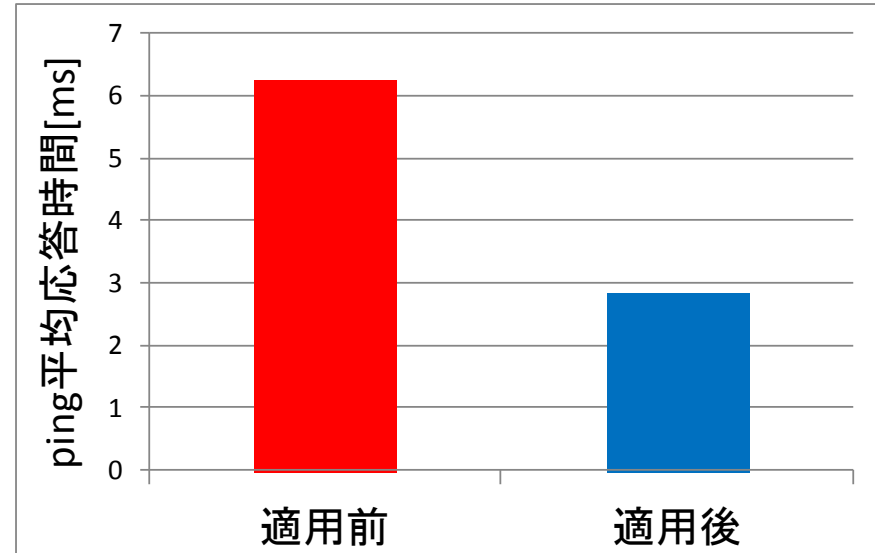
- OVCT適用前のOpenFlowネットワーク

通信速度

15/18

- TCLとVMを各2台ずつ用意し、それぞれのping応答時間を計測

(単位:ms)		宛先			
送信元		適用前		適用後	
		VM1	VM2	VM1	VM2
	TCL1	6.02	5.59	2.77	2.75
	TCL2	6.88	6.55	2.98	2.83
平均		6.45	6.07	2.88	2.79
		6.26		2.84	

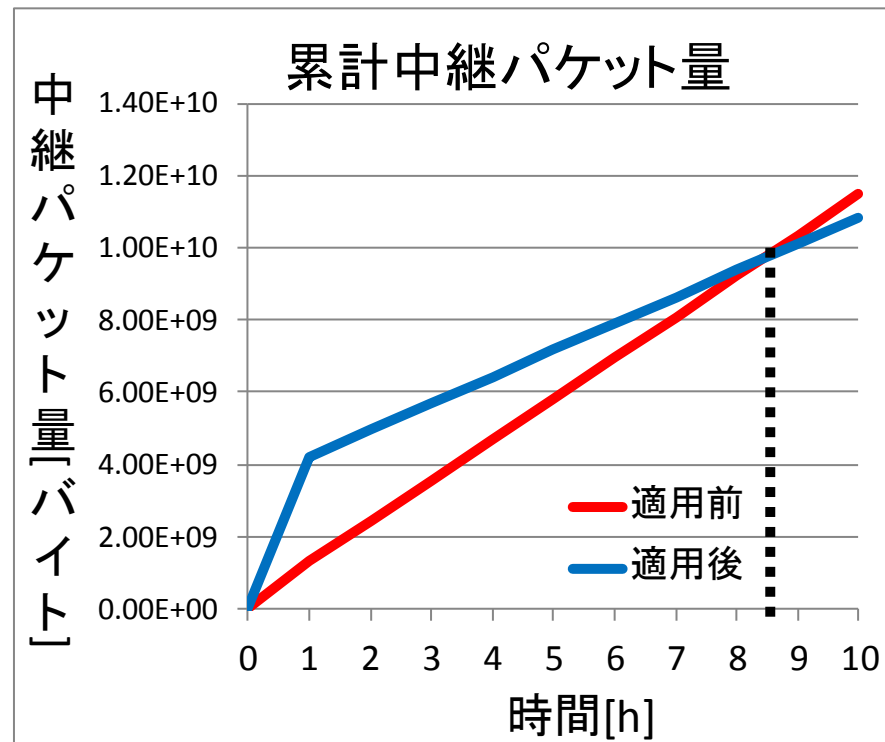
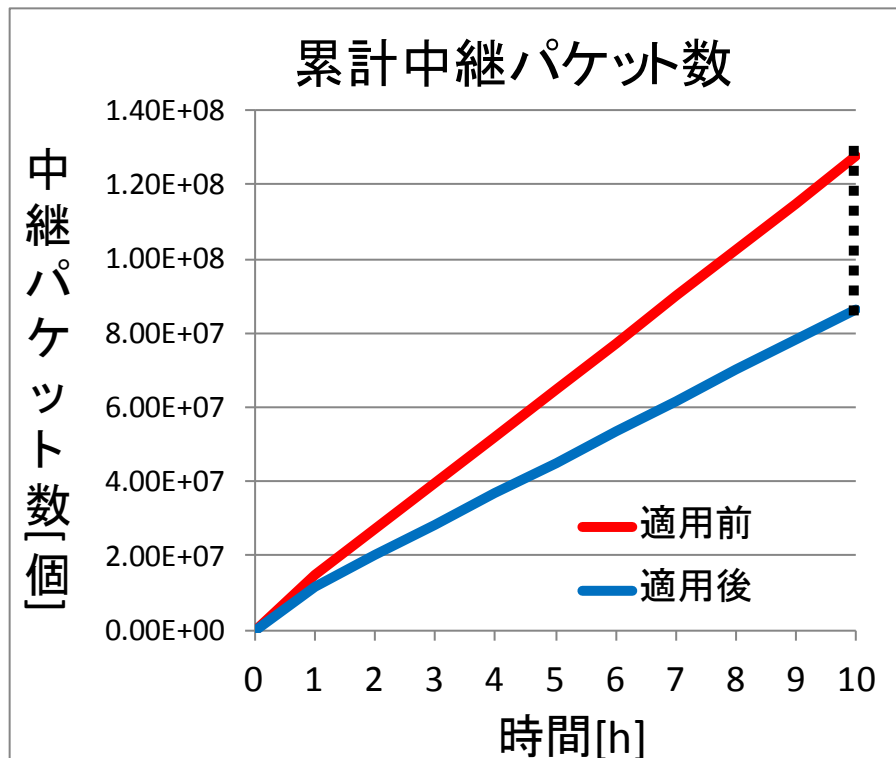


OVCT適用前と比較し, 55%向上
TCLシステムの性能向上

中継パケット数・量

16/18

- ネットワーク全体のスイッチが転送したパケットを測定



パケット数を最大32%削減

9時間以上稼働させることでパケット量を軽減

- 仮想マシン型のTCLシステム

- 入出力がネットワークを介するため、性能が低下する

LMとOpenFlowを用いた
仮想マシンの配置システムを提案, 実装

- TCLの位置に応じて仮想マシンを配置

- 配置場所へLML, OpenFlowでネットワーク設定を自動変更

端末-仮想マシン間の通信速度を55%向上
中継パケット数を最大32%削減
9時間以上稼働させることでパケット量を軽減

- 今後の課題

- 仮想マシンの配置場所の最適化
 - 実用的なネットワーク構成への対応

最後に、(日工専)、(1サ本)の方々をはじめ、この1年間私を助けて下さった全ての方々に、この場を借りまして心から深く感謝申し上げます。