

Design, Modeling, and Evaluation of Efficient Caching Mechanisms for Content Dissemination Networks

コンテンツ流通ネットワークにおける効果的なキャッシング機構の設計、モデル化、評価に関する研究

大阪大学大学院情報科学研究科
情報ネットワーク学専攻
村田研究室
今井 悟史

Advanced Network Architecture Research Group
<http://www.anarg.jp/>

1

研究背景

- ICTサービスの普及→ 通信トラフィック増加
- キャッシング技術による効率的なデータ配信に期待
 - CDN/エッジコンピューティングから、ICNに展開

CDN : Contents Delivery Network, ICN : Information-Centric Networking

2

代表的なICNアーキテクチャ

- CCN(Xerox PARC, 2007)、NDN(UCLA, 2010)等
 - コンテンツ配布・発見・配信・蓄積を各ノードが自律分散処理
 - コンテンツ名によるルーティング、配信木 (キャッシング階層) の自動生成、多重化

キャッシング階層およびアクセスパターンの不均一化、メモリおよびネットワーク帯域の共有

CCN: Content-Centric Networking, NDN : Named Data Networking

3

関連技術動向とICNの設計要件

- 関連技術
 - 機能・リソースの利用粒度を柔軟に変更・管理するネットワーク技術
 - Energy Proportional Network (省電力ネットワーク)
 - NFV/SDN (仮想化ネットワーク)
 - 実際のアクセスパターンやコンテンツ特性に適した効率的なキャッシング技術
 - One-Timers(1回だけアクセスされるコンテンツ)管理
 - Time-To-Live (キャッシング保持時間)管理
- 設計要件
 - 省電力化: 消費電力量を削減するキャッシング管理方法の実現
 - 解析性の向上: キャッシュ方法、メモリサイズ、キャッシング性能(ヒット率、経路長)、キャッシング階層構造、ネットワーク帯域との関係明確化
 - 適応性・制御性の向上: リクエスト変化に応じた、動的なキャッシング制御方法の実現

NFV : Network Functions Virtualization
SDN : Software Defined Network

4

研究目的

ICNの設計要件	研究内容
省電力化	省電力なキャッシング配置設計方法、および分散キャッシング判定方法の提案 (Chapter2)
解析性の向上	One-Timersに適したノードレベルキャッシング機構に対する解析モデルの提案 (Chapter3)
	Time-To-Live(TTL)による階層型キャッシングネットワークに対する解析モデルの提案 (Chapter4)
適応性・制御性の向上	Time-To-Live(TTL)による適応型リソース制御方法の提案 (Chapter5)

5

博士論文の構成

- Chapter 1: Introduction
- Chapter 2: Design, Modeling, and Evaluation of Energy Efficient Cache Management
 - 省電力なキャッシング配置設計、および分散キャッシング判定方法の設計、モデル化、評価
- Chapter 3: Modeling and Evaluation of Node-Level Cache Management
 - One-Timersに適したノードレベルキャッシング機構に対する解析モデルの構築、評価
- Chapter 4: Modeling and Evaluation of Static TTL Management in Hierarchical Caching
 - Time-To-Live(TTL)による階層型キャッシングネットワークに対する解析モデルの構築、評価
- Chapter 5: Design, Modeling, and Evaluation of Adaptive TTL Management in Hierarchical Caching
 - Time-To-Live(TTL) による適応型リソース制御方法の設計、モデル化、評価
- Chapter 6: Conclusion

6

Chapter 2: Design, Modeling, and Evaluation of Energy Efficient Cache Management

省電力なキャッシュ配置設計、および分散キャッシュ判定方法の設計、モデル化、評価

1. Satoshi Imai, Kenji Leibnitz, Masayuki Murata, "Energy Efficient Data Caching for Content Dissemination Networks", *Journal of High Speed Networks*, DOI:10.3233/JHS-130474, vol.19, no. 3, pp. 215-235, October, 2013.
2. Satoshi Imai, Kenji Leibnitz, Masayuki Murata, "Energy Efficient Content Locations for In-Network Caching", in *Proceedings of 18th Asia-Pacific Conference on Communications (APCC 2012)*, Jeju, Korea, Oct. 2012.
3. Satoshi Imai, Kenji Leibnitz, Masayuki Murata, "Energy-Aware Cache Management for Content-Centric Networking", in *Proceedings of First International Workshop on Energy-Aware Systems, Communications and Security*, Barcelona, Spain, March 2013.

7

Chapter2の研究背景と課題

- トラフィック増加に伴い、ネットワークの消費電力量が増大

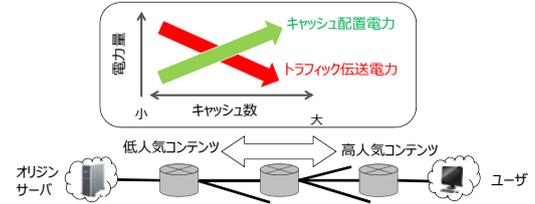
→ キャッシュによるトラフィック量削減、省電力化に期待

- 省電力デバイス (Sleep/DVS等) の普及

→ Energy Proportional Network^[21]の実現

DVS: Dynamic Voltage Scaling

[課題] キャッシュ階層と電力トレードオフを考慮したコンテンツ配置



8

[21] P. Mahadevan, P. Sharma, S. Banerjee, P. Ranganathan, "A power benchmarking framework for network devices," in *Proceedings of NETWORKING'09*, 2009

Chapter2のアプローチ

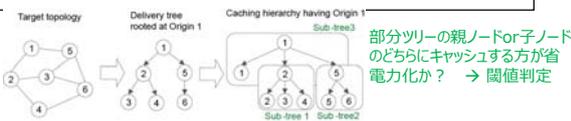
- 従来のコンテンツ配置問題^[38,39,54]に対し、キャッシュ階層の多重化と電力モデルの考慮を加え、新たな設計問題として定式化、及びヒューリスティックな分散アルゴリズムを提案

全域情報を用いた最適キャッシュ配置方法

0-1整数計画法を用いて、省電力なキャッシュ配置箇所を設計

局所情報を用いた分散キャッシュ判定方法

部分ツリーの情報からリクエストに対する閾値を導出し、各ノードにおいて、閾値によるキャッシュ判定を実施



[38] U. Lee, I. Rimac, D. C. Kilper, V. Hilt, "Toward Energy-Efficient Content Dissemination," *IEEE Network*, 2011.
 [40] K. Guan, G. Atkinson, D. C. Kilper, "On the Energy Efficiency of Content Delivery Architectures," in *Proceedings of the 4th IEEE International Conference on Communications (ICC) Workshop on Green Communications*, 2011.
 [54] L. Qiu, V. N. Padmanabhan, G. M. Voelker, "On the Placement of Web Server Replicas," in *Proceedings of INFOCOM*, 2001.

9

提案方法

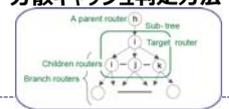
- 最適キャッシュ配置方法

目的関数

$$\text{Minimize } \sum_{i \in V} V(C_{0i} \cdot u_i) + \sum_{(i,j) \in E} \{T_{k(i,j)} \cdot s_{(i,j)}\}$$

キャッシュ配置電力

トラフィック伝送電力



$$Power_h^i(\lambda_h^i) \ge or < \sum_{j \in c_i} Power_j^j(\lambda_h^j)$$

親ノードにキャッシュor 子ノード群にキャッシュした場合の電力関係をリクエストレートに対する線形不等式でモデル化、各ノードのレート閾値を導出

$$\lambda_h^i = \frac{(Cr_h^i - 1) \cdot D \cdot P_{con}}{D \cdot (P_r + P_{con}) + (H_h^i - \sum_{j \in c_i} H_j^j)}$$

Threshold-based Caching

閾値を上回る: 自ノードにキャッシュ
 閾値を下回る: 上位ノードに転送

キャッシュ階層に基づく制約

$$\text{subject to}$$

$$\sum_j s_{(i,j)} = 1 \quad \forall i$$

$$s_{(i,i)} \leq u_i \quad \forall i, j$$

$$s_{(i,j)} = 0 \quad \forall s_{(h,j)} \notin s_{(i,j)}^*$$

$$s_{(i,j)} + u_k \leq 1 \quad \forall i, j, k \in \{s_{(i,j)}^* - i\}$$

10

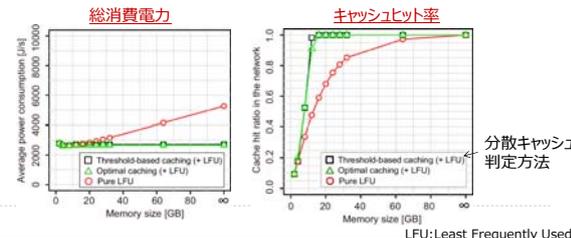
省電力効果とキャッシュヒット率(シミュレーション)

- 分散キャッシュ判定方法により、最適キャッシュ配置方法と同等の電力量、キャッシュヒット率を実現
- 従来手法 (LFU) よりも省電力、かつキャッシュヒット率を向上可能

リクエスト分布: Zipf (1.2) 評価トポロジ
 オブジェクト数: 10000
 オリジンサイト: 一様分布
 電力パラメータ

Device (Product)	Power	Spec	Power Density
TERAM	29W	4GB	$P_{con} = 7.25 \times 10^{-3} [W/GB]$
Colson Router (R8-1)	415W	320Gbps	$P_{con} = 1.3 \times 10^{-3} [W]$
WDMPLASERWOW15000	800W	480Gbps	$P_{con} = 1.67 \times 10^{-3} [W]$

省電力化とキャッシュ性能向上を両立



11

Chapter 3: Modeling and Evaluation of Node-Level Cache Management

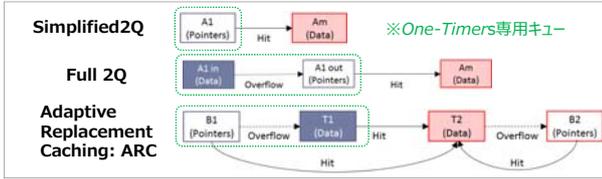
One-Timersに適したノードレベルキャッシュ機構に対する解析モデルの構築、評価

1. Satoshi Imai, Kenji Leibnitz, Masayuki Murata, "Statistical Approximation of Efficient Caching Mechanisms for One-Timers", submitted.

12

Chapter3の研究背景と課題

- VoDやWebは、低頻度アクセスのOne-Timers^[24,26]が多数
 - LRU/LFUでは、One-Timersの影響で、単一ノードでのキャッシュ性能が劣化
 - キュー分割による2Q^[13]やARC^[14]が有効 (ただし、これらの技術は、実装論の議論がメインで、そのキャッシュ特性は未だ理論的に解析されていない)



[課題] 単一ノードでの2Q・ARC機構における理論的な性能解析

[13] T. Johnson, D. Shasha, "2Q: A Low Overhead High Performance Buffer Management, Replacement Algorithm.", 1994.
 [14] N. Megiddo, D.S. Modha, "ARC: A Self-Tuning, Low Overhead Replacement Cache," in Proceedings of FAST, 2003.
 [24] N. Arlitt, C. Williamson, "Internet Web Servers: Workload Characterization and Performance Implications," IEEE/ACM Transaction on Networking, 1997.
 [26] P. Gill, M. Arlitt, Z. Li, "Youtube Traffic Characterization: a View from the Edge," in Proceedings of IMC'07, 2007.

Chapter3のアプローチ

- LRU近似モデル(Che Approximation)^[11,34]を参考にし、2QおよびARCの各キューの相互影響を考慮して拡張モデル化

Che Approximation

アクセス頻度 $\lambda(n)$ を持つオブジェクトnに対するキャッシュヒット率:

$$H_{LRU}(n) = 1 - e^{-\lambda(n)\tau_C}$$

特性時間 τ_C は、メモリサイズCに対し、以下の条件を満たすように算出

$$C = \sum_{n=1}^N 1 - e^{-\lambda(n)\tau_C}$$

キューサイズとキャッシュヒット率の関係をモデル化

[11] H. Che, Y. Tung, Z. Wang, "Hierarchical Web Caching Systems: Modeling, Design and Experimental Results," IEEE Journal on Selected Areas in Communications, 2002.
 [34] C. Fricker, P. Robert, J. Roberts, "A Versatile and Accurate Approximation for LRU Cache Performance," in Proceedings of the 24th International Teletraffic Congress (ITC), Krakow, Poland, 2012.

提案モデル

- Simplified 2Q-Full 2Q (分割サイズ固定)、ARC (分割サイズ動的調整) に対し、近似モデルを構築

ARC行々

One-Timers用のキューサイズ(p)が動的に変更

Zipfアクセスパターンに対しpはメモリサイズの1%程度(微小値)で変動

pを固定値で扱う Fixed Replacement Caching: FRC(p=1)で近似

FRC(p) Model for ARC

アクセス頻度 $\lambda(n)$ を持つオブジェクトnに対するキャッシュヒット率 $H_{FRC}(n) = H_{T1}(n) + H_{T2}(n)$

各キューでのヒット率およびアクセス確率

$$H_{T1}(n) = 1 - e^{-\lambda_{T1}(n)\tau_{T1}}$$

$$H_{T2}(n) = 1 - e^{-\lambda_{T2}(n)\tau_{T2}}$$

$$\lambda_{T1}(n) = \lambda(n)H_{T2}(n) + \lambda_{T1}(n)H_{T1}(n) + \lambda_{B1}(n)H_{B1}(n) + \lambda_{B2}(n)H_{B2}(n)$$

$$\lambda_{T2}(n) = \lambda(n)(1 - H_{T1}(n))(1 - H_{B1}(n))$$

$$\lambda_{B1}(n) = \frac{1 - (1 - \lambda_{T1}(n))^{\tau_{T1}}}{\tau_{T1}}$$

$$\lambda_{B2}(n) = \frac{1 - (1 - \lambda_{T2}(n))^{\tau_{T2}}}{\tau_{T2}}$$

$$\lambda_{T1}(n) = \lambda(n)(1 - H_{T1}(n))(1 - H_{B1}(n))$$

$$\lambda_{T2}(n) = \lambda(n)(1 - H_{T1}(n))(1 - H_{B1}(n))(1 - H_{B2}(n))$$

各キューの特性時間 $\tau_{T2}, \tau_{T1}, \tau_{B2}, \tau_{B1}$ は、各キュー長条件を満たすように算出

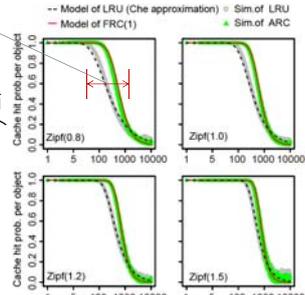
$$p = \sum_{n=1}^N H_{T1}(n), \quad C - p = \sum_{n=1}^N H_{T2}(n), \quad C - p = \sum_{n=1}^N H_{B1}(n), \quad p = \sum_{n=1}^N H_{B2}(n).$$

ARCモデルの近似性能(シミュレーションvsモデル)

- 提案モデルにより、シミュレーションデータとほぼ同等のキャッシュヒット率を推定

メモリサイズ: 1000
 オブジェクト数: 10000
 オブジェクトサイズ: 1

- ARCは、LRUよりも、Warmオブジェクトのキャッシュヒット率が高く、提案モデルによりその傾向を高精度に推定
- 提案モデルにより、シミュレーション評価よりも、高速かつ簡単に、各オブジェクトのキャッシュヒット率を評価可能



2QやARCに基づくキャッシュ機構の設計に活用

Chapter 4: Modeling and Evaluation of Static TTL Management in Hierarchical Caching

Time-To-Live (TTL)による階層型キャッシュネットワークに対する解析モデルの構築、評価

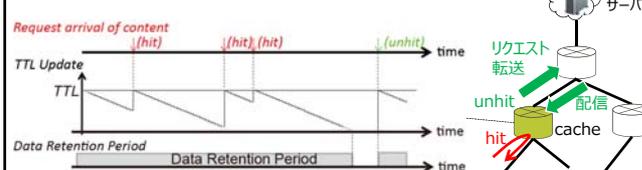
- Satoshi Imai, Kenji Leibnitz, Masayuki Murata, "Modeling of Content Dissemination Networks on Multiplexed Caching Hierarchies," in Proceedings of the 13th International Conference on Networks, Nice, France, 23-27 Feb. 2014.

Chapter4の研究背景と課題

- メモリ利用効率を向上させる上で、キャッシュ保持時間(TTL)を用いた管理も有効

- LRU/LFU/2Q/ARC等のメモリ量をすべて使い切る事を前提としたキャッシュ技術と異なり、オブジェクト毎のきめ細かいキャッシュ管理が可能

代表的なTTL管理方法 (タイムアウトしたらキャッシュ廃棄)



[課題] 階層型キャッシュネットワークに対し、TTL管理を適用した場合のリソース利用、キャッシュ性能への影響把握

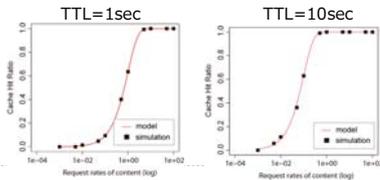
Chapter4のアプローチ

- 単一ノードのTTLモデルを連結して、ネットワーク全体のキャッシュ特性をモデル化

単一ノードレベルのTTLモデル

アクセス頻度 $\lambda(n)$ を持つオブジェクト n に対し、保持時間 TTL を設定したときのキャッシュヒット率:

$$f(\lambda^c, TTL^c) = 1 - e^{-\lambda^c TTL^c}$$



▶ 19

提案モデル

- 階層型キャッシュネットワークに対するリクエスト伝搬モデル

$$\Lambda^c[s+1] = D^c[s] \cdot \Lambda^c[s] + R^c, \forall c \quad (\text{行列式})$$

ノード i からノード j へのリクエストレート 各拠点からのリクエスト発生量

$$\Lambda^c[s] := \begin{bmatrix} \lambda_{(L,1)}^c[s] & \dots & \lambda_{(L,N)}^c[s] \\ \vdots & \ddots & \vdots \\ \lambda_{(M,1)}^c[s] & \dots & \lambda_{(M,N)}^c[s] \end{bmatrix}$$

$$[R^c]_{ij} := \begin{cases} r_j^c, & \text{when } CR_i \text{ is located on } j\text{-th request site} \\ 0 & \text{otherwise} \end{cases}$$

配信木に応じた確率伝搬行列

$$D^c[s] := \begin{cases} 1 - f(\sum_{n=1}^N \lambda_{(m,n)}^c[s], TTL^c), & \forall m = \text{parent_node}(n) \\ 0 & \text{otherwise} \end{cases}$$

- リソース利用量・品質モデル

- ✓ メモリ使用量
- ✓ データ伝送量
- ✓ キャッシュヒット率
- ✓ 配信経路長
- ✓ 消費電力量

$$\Lambda^c[s] = \begin{bmatrix} \lambda_{(1,1)}^c[s] & \lambda_{(1,2)}^c[s] & \lambda_{(1,3)}^c[s] & \lambda_{(1,4)}^c[s] & \lambda_{(1,5)}^c[s] \\ \lambda_{(2,1)}^c[s] & \lambda_{(2,2)}^c[s] & \lambda_{(2,3)}^c[s] & \lambda_{(2,4)}^c[s] & \lambda_{(2,5)}^c[s] \\ \lambda_{(3,1)}^c[s] & \lambda_{(3,2)}^c[s] & \lambda_{(3,3)}^c[s] & \lambda_{(3,4)}^c[s] & \lambda_{(3,5)}^c[s] \\ \lambda_{(4,1)}^c[s] & \lambda_{(4,2)}^c[s] & \lambda_{(4,3)}^c[s] & \lambda_{(4,4)}^c[s] & \lambda_{(4,5)}^c[s] \\ \lambda_{(5,1)}^c[s] & \lambda_{(5,2)}^c[s] & \lambda_{(5,3)}^c[s] & \lambda_{(5,4)}^c[s] & \lambda_{(5,5)}^c[s] \end{bmatrix}$$

$$R^c = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ r_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$D^c[s] = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & e^{-\sum_{n=1}^N \lambda_{(2,n)}^c[s] TTL^c} & 0 & 0 & 0 \\ 0 & 0 & e^{-\sum_{n=1}^N \lambda_{(3,n)}^c[s] TTL^c} & 0 & 0 \\ 0 & 0 & 0 & e^{-\sum_{n=1}^N \lambda_{(4,n)}^c[s] TTL^c} & 0 \\ 0 & 0 & 0 & 0 & e^{-\sum_{n=1}^N \lambda_{(5,n)}^c[s] TTL^c} \end{bmatrix}$$

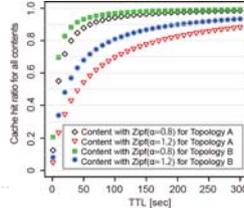
▶ 20

TTL値によるリソース・品質影響解析 (提案モデルによる評価)

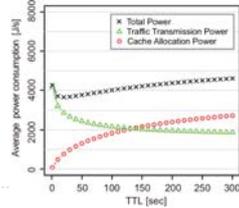
- TTL値によるリソース利用量 (メモリ、帯域、消費電力量など)、品質影響を容易かつ高速に評価可能

TTL値に基づく階層型キャッシュネットワークの設計に活用

キャッシュヒット率



総消費電力量



▶ 21

Chapter 5: Design, Modeling, and Evaluation of Adaptive TTL Management in Hierarchical Caching

Time-To-Live (TTL) による適応型リソース制御方法の設計、モデル化、評価

- Satoshi Imai, Kenji Leibnitz, Masayuki Murata, "Adaptive TTL Control to Minimize Resource Cost in Hierarchical Caching Networks", to appear in IEICE Transactions on Information and Systems, March, 2015.

▶ 22

Chapter5の研究背景と課題

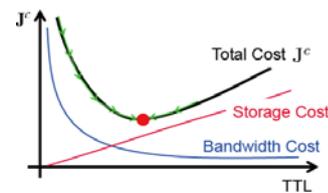
- 仮想ネットワーク上で、ICNサービスを展開
 - ストレージ、データ伝送に伴う帯域など、所要リソースに対する細かなコスト定義が可能 (電力コストの一つ)
- TTL値を動的に変更することにより、コンテンツオブジェクト毎のリソースコストを制御可能

[課題] TTL値、リクエスト分布、キャッシュ階層に影響されるリソースコストの一元管理および制御

▶ 23

Chapter5のアプローチ

- 現在のTTL値およびリクエストレート計測値からリソースコストを予測、その微分 (傾き) を用いて、コストが減少するようにTTL値を制御



TTL制御器

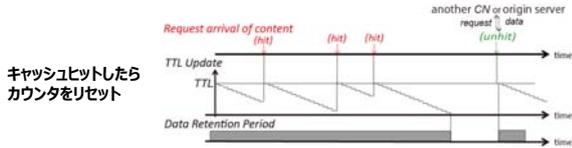
$$TTL_{k+1}^c = \Theta(TTL_k^c - \text{sgn}(dJ^c/dTTL) \delta t_c)$$

TTL値を δt_c 変化した時のコストの傾きを予測、コストが減少するようにTTLを更新

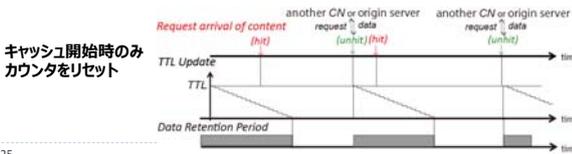
▶ 24

想定するTTL管理方法、TTLモデル(2種類)

Policy1 $f(\lambda^c, TTL^c) = 1 - e^{-\lambda^c TTL^c}$



Policy2 $f(\lambda^c, TTL^c) = \frac{\lambda^c TTL^c}{1 + \lambda^c TTL^c}$



▶ 25

提案方法 (1/2)

リソースコストの予測モデル

✓ トータルリソースコスト ← 非線形モデル

$$J^c(TTL^c, \lambda_{(m,n)}^c) = \theta_c C_s [1, \dots, 1]_{1 \times M} \begin{bmatrix} f(\sum_j \lambda_{(1,j)}^c, TTL^c) \\ \vdots \\ f(\sum_j \lambda_{(M,j)}^c, TTL^c) \end{bmatrix} + \theta_c C_b [1, \dots, 1]_{1 \times N} \begin{bmatrix} Tr_1^c \\ \vdots \\ Tr_N^c \end{bmatrix}$$

ストレージコスト 帯域コスト

✓ リソースコストの微分(線形化)

$$\delta J^c = \frac{\partial J^c}{\partial TTL} \delta TTL^c + \sum_{(m,n)} \frac{\partial J^c}{\partial \lambda_{(m,n)}^c} \delta \lambda_{(m,n)}^c = Q^c \delta TTL^c \quad \text{TTLの更新量 } \delta TTL^c \text{ の線形関数として導出可能}$$

TTL制御器

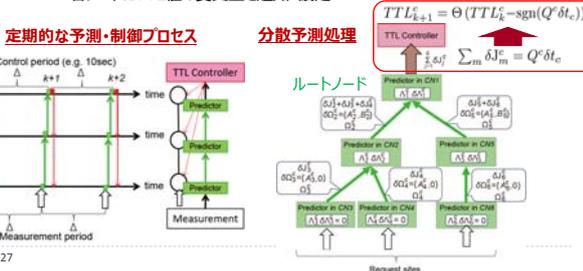
$$TTL_{k+1}^c = \Theta(TTL_k^c - \text{sgn}(dJ^c/dTTL) \delta TTL^c) = \Theta(TTL_k^c - \text{sgn}(Q^c \delta TTL^c))$$

▶ 26

提案方法 (2/2)

分散制御アルゴリズム $\delta J^c = \sum_m \delta J_m^c$ 各ノードのコストに分割

- ✓ 各ノードで各オブジェクトのリクエストレートを定期計測
- ✓ 自ノードのコスト情報 δJ_m^c と、親ノードへのリクエスト伝搬量の予測し、親ノードに伝搬
- ✓ ルートノードで、収集したコスト予測値を集計し、傾きからTTL値の増減を決定 → 各ノードにTTL値の変更量を通知・設定



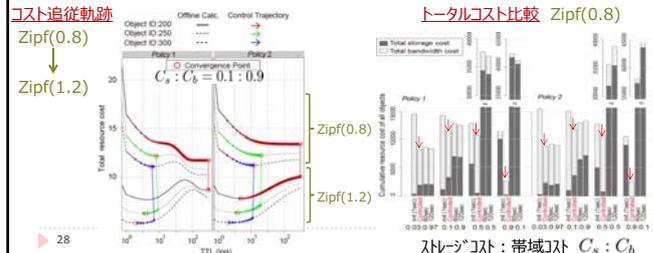
▶ 27

TTL制御効果

- リクエスト分布変化に応じて、予測コストが極小になるようにTTL値を制御可能
- 静的にTTL値を設定した場合と比較し、リソースコストを削減可能



TTLによるキャッシュネットワークの効率的なリソース制御を実現



▶ 28

本論文のまとめ

まとめ

- ✓ 省電力化技術
 - 最適キャッシュ配置設計方法と分散キャッシュ判定方法を提案
- ✓ 解析技術
 - One-Timersに適したノードレベルキャッシュ機構に対する解析モデルを提案
 - TTLによる階層型キャッシュネットワークに対する解析モデルを提案
- ✓ 適応制御技術
 - TTLによる予測モデルを用いたリソース制御方法を提案

今後の課題

- ✓ リソース利用効率とサービス品質(遅延等)のトレードオフ考慮
- ✓ 非定常なアクセス特性の考慮
- ✓ スケーラビリティ向上

▶ 29

ご清聴ありがとうございました。

▶ 30