Topic-based Delivery of Event Messages in Peer-to-Peer Publish/Subscribe Systems

Nakayama, Hiroki

(出版者 / Publisher)
法政大学大学院理工学・工学研究科
(雑誌名 / Journal or Publication Title)
法政大学大学院紀要. 理工学・工学研究科編 / 法政大学大学院紀要. 理工学・工学研究科編
(巻 / Volume)
58
(開始ページ / Start Page)
1
(終了ページ / End Page)
4
(発行年 / Year)
2017-03-31
(URL)
https://doi.org/10.15002/00014349

Topic-based Delivery of Event Messages in Peer-to-Peer Publish/Subscribe Systems

Hiroki Nakayama
Supervisor : Professor Makoto Takizawa
Graduate School of Science and Engineering, Hosei University

An event-driven publish/subscribe (PS) model of a distributed system is used in various applications. In this paper, we discuss a peer-to-peer (P2P) model of a topic-based PS system (P2PPS model) where each peer process (peer) can both subscribe interesting topics and publish event messages with topics. In our previous studies, we propose the TBC (topic-based-causally delivery) protocol a homogeneous networks where maximum delay time between every pair of peers is same. Here, a pair of event messages are checked if the event messages are related in the topic vector and are causally delivered by taking advantage of physical time and linear time. In this paper, we consider a system where peers are interconnected in heterogeneous networks. Here, maximum delay time between every pair of peers is not same. We propose a heterogeneous TBC (HTBC) protocol where event messages are TBC-causally delivered to target peers in heterogeneous network. We evaluate the HTBC protocol and show the number of pair of event messages unnecessarily ordered is reduced in the HTBC protocol.

Key Words *: Publish/subscribe (PS) systems, Peer-to-peer (P2P) model, Topic-based causal dependency, Topic vectors, Heterogeneous TBC (HTBC) protocol, Linear clock, Physical clock, Heterogeneous networks;*

1. INTRODUCTION

In distributed applications, a group of processes $p_1, \ldots, p_n$ $(n \geq 1)$ are cooperating with one another by exchanging messages in underlying networks. A process is modeled to be a finite state machine. Event-driven distributed systems like publish/subscribe (PS) [8], [9] systems are developed and used in various types of distributed applications like Google alert. A peer-to-peer (P2P) publish/subscribe (P2PPS) [3], [4], [5], [6] system is composed of peer processes (peers) where each peer can play both subscriber and publisher roles. In this paper, we consider a heterogeneous system where the maximum clock offset of each peer is the same but the maximum delay time between every pair of peers is not the same. Peers in each subgroup are interconnected in a local area network (LAN) and subgroups are interconnected in wide area network (WAN).

In this paper, we newly propose a heterogeneous topic-based-causally delivery (HTBC) protocol to topic-based-causally (TBC) deliver event messages by taking advantage of linear clock and physical clock. Event messages are required to be causally delivered to every common target peer in a group of multiple peers. An event message e_1 topic-based-causally (TBC) precedes an event message e_2 in a heterogeneous network if and only if (iff) not only e_1 causally precedes e_2 $(e_1 \rightarrow e_2)$ but also e_1 and e_2 are related in terms of topics [7].

In section 2, we propose the HTBC protocol to topic-based-causally deliver event messages in heterogeneous networks. In section 3, we evaluate the HTBC protocol in terms of number of pairs of event messages unnecessarily ordered.

2. HTBC PROTOCOL

We propose a heterogeneous TBC (HTBC) protocol to topic-based-causally (TBC) deliver event messages in heterogeneous network. We discuss how each peer p_i orders event messages which are notified to the peer p_i in the HTBC protocol. A pair of event messages e_1 and e_2 are ordered in the following rules :

[Ordering rules] Let e_1 and e_2 be a pair of event messages notified to a peer p_i. The event messages e_1 and e_2 are ordered as follows :

1) e_1 *TV - precedes* e_2 $(e_1 \Rightarrow_{TV} e_2)$ if $e_1.TV \prec_{e_1.P \cap e_2.P \cap p_i.S} e_2.TV$.

2) e_1 *LT-precedes* e_2 $(e_1 \Rightarrow_{LT} e_2)$ if $e_1.LT \leq e_2.LT$.

3) e_1 *PT - precedes* e_2 $(e_1 \Rightarrow_{PT} e_2)$ if $e_1.PT - e_2.PT < 2\varepsilon$ and one of the following conditions holds :

a) $e_2.PT - e_1.PT > mxd_{ij} + 2\varepsilon$.

b) If $mind_{ij} \geq 2\varepsilon$, $mind_{ij} - 2\varepsilon \leq e_2.PT - e_1.PT \leq maxd_{ij} + 2\varepsilon$.

4) e_1 TBC-$precedes$ e_2 ($e_1 \Rightarrow_{TBC} e_2$) if $e_1 \Rightarrow_{LT} e_2$, $e_1 \Rightarrow_{TV} e_2$, and $e_1 \Rightarrow_{PT} e_2$.

We consider the following types of protocols based on the ordering rules :

- TV protocol: an event message e_1 precedes an event message e_2 if $e_1 \Rightarrow_{TV} e_2$.
- TV_LT protocol: an event message e_1 precedes an event message e_2 ($e_1 \Rightarrow_{TV\_LT} e_2$) if $e_1 \Rightarrow_{TV} e_2$ and $e_1 \Rightarrow_{LT} e_2$.
- TV_PT protocol: an event message e_1 precedes an event message e_2 ($e_1 \Rightarrow_{TV\_PT} e_2$) if $e_1 \Rightarrow_{TV} e_2$ and $e_1 \Rightarrow_{PT} e_2$.
- HTBC protocol: an event message e_1 precedes an event message e_2 ($e_1 \Rightarrow_{TBC} e_2$) if $e_1 \Rightarrow_{TBC} e_2$, i.e. $e_1 \Rightarrow_{TV} e_2$, $e_1 \Rightarrow_{LT} e_2$, and $e_1 \Rightarrow_{PT} e_2$.

3. EVALUATION

We evaluate the TV, TV_ LT, TV_ PT, and HTBC protocols in terms of number of pairs of event messages unnecessarily ordered. A system S is composed of n ($\geq$ 1) peers $p_1, \dots, p_n$. A pair of peers p_i and p_j are interconnected in a local area network (LAN) and wide area network (WAN), where $maxd_{ij} = 16$ [time unit (tu)] and $maxd_{ij} = 160$ [tu], respectively. The maximum clock offset ε is same in every peer, $\varepsilon = 4$ [tu].

A topic set TT is composed of h ($\geq$ 1) topics $t_1, \dots, t_h$. A subscription $p_i.S$ of a peer p_i includes h_i ($h_i \leq h$) topics ($p_i.S \subseteq TT$). A publication $e.P$ of an event message e published by a peer p_i includes k_i ($k_i \leq h$) topics ($e.P \subseteq TT$). Let xt show the maximum number of topics to be included in each of the subscription $p_i.S$ of a peer p_i and publication $e.P$ of an event message e published by the peer p_i ($xt \leq h$). Here, the numbers h_i and k_i of topics are randomly taken out of 1 to xt for each subscription $p_i.S$ and each event message which p_i publishes, respectively, $1 \leq h_i \leq xt$ and $1 \leq k_i \leq xt$.

Each peer p_i randomly publishes event messages. Total number m of event messages are randomly published by peers in a system. For each event message e, a source peer p_i which publishes the event message e is randomly selected. In addition, the publishing time τ_i when a peer p_i publishes an event message e is randomly taken from 0 to $maxT - 1$ [tu]. The simulation time $maxT$ is 150,000 [tu]. Suppose a peer p_i publishes an event message e at time τ_i. The event message e is delivered to every peer p_j at time τ_j. Here, $\tau_j = \tau_i + \delta_{ij}$. The delay time δ_{ij} between a pair of peers p_i and p_j is randomly decided between the minimum delay time $mind$ and the maximum delay time $maxd$, $mind \leq \delta_{ij} \leq maxd$.

An event message e also carries the vector time $e.VT$ to check the causal precedency of event messages in the evaluation. The vector time $VT = \langle vt_1, \dots, vt_n \rangle$ is manipulated by each peer p_i ($i = 1, \dots, n$) in the vector clock (VT) protocol [2] as follows :

1) [**Initially**] In every peer p_i, $VT = \langle 0, \dots, 0 \rangle$;
2) [**Receive**] On receiving an event message e, a peer p_i manipulates the vector time VT as follows :
   $vt_i = vt_i + 1$ for $j = 1, \dots, n$ ($j \neq i$);
3) [**Publish**] A peer p_i would like to publish an event message e;
   $vt_i = vt_i + 1$;
   $e.VT = VT$;
   p_i publishes e;

Here, an event message e_1 causally precedes an event message e_2 ($e_1 \rightarrow e_2$) [1] iff $e_1.VT \leq e_2.VT$. It is checked if an event message e_1 causally precedes an event message e_2 by using the vector time $e_1.VT$ and $e_2.VT$. Here, $e_1 \Rightarrow_{VT} e_2$ iff $e_1.VT \leq e_2.VT$.

In the simulation, we collect the following ordered pairs of event messages which each peer pi receives :

1) TV_i is a set of pairs of event messages e_1 and e_2 which a peer p_i orders as $e_1 \Rightarrow_{TV} e_2$ only in the topic vector (TV) protocol, i.e. $TV_i = \{\langle e_1, e_2 \rangle \mid e_1.TV \leq e_2.TV\}$.
2) TV_LT_i is a set of event messages e_1 and e_2 which a peer p_i orders as $e_1 \Rightarrow_{TV} e_2$ and $e_1 \Rightarrow_{LT} e_2$ in the TV_LT protocol, i.e. $TV\_LT_i = \{\langle e_1, e_2 \rangle \mid e_1 \Rightarrow_{TV} e_2$ and $e_1 \Rightarrow_{LT} e_2$ $(e_1.LT \leq e_2.LT)\}$.
3) TV_PT_i is a set of event messages e_1 and e_2 which a peer p_i orders as $e_1 \Rightarrow_{TV} e_2$ and $e_1 \Rightarrow_{PT} e_2$ in the TV_PT protocol, i.e. $TV\_PT_i = \{\langle e_1, e_2 \rangle \mid e_1 \Rightarrow_{TV} e_2$ and $e_1 \Rightarrow_{PT} e_2$ (i.e. $e_2.PT - e_1.PT > maxd + 2\varepsilon$ or if $mind \geq 2\varepsilon, mind - 2\varepsilon \leq e_2.PT - e_1.PT \leq maxd + 2\varepsilon$)$\}$.
4) TV_VT_i is a set of event messages e_1 and e_2 which are ordered as $e_1 \Rightarrow_{TV} e_2$ and $e_1 \Rightarrow_{VT} e_2$, i.e $TV\_VT_i = \{\langle e_1, e_2 \rangle \mid e_1.TV \leq e_2.TV$ and $e_1.VT \leq e_2.VT\}$.
5) $HTBC_i$ is a set of pairs of event messages e_1 and e_2 which a peer pi orders as $e_1 \Rightarrow_{TV} e_2$, $e_1 \Rightarrow_{LT} e_2$, and $e_1 \Rightarrow_{PT} e_2$ in the HTBC protocol, i.e. $HTBC_i = \{\langle e_1, e_2 \rangle \mid e_1 \Rightarrow_{HTBC} e_2$, i.e. $e_1 \Rightarrow_{TV} e_2, e_1 \Rightarrow_{LT} e_2$ and $e_1 \Rightarrow_{PT} e_2\}$.
6) VT_i is a set of pairs of event messages e_1 and e_2 where e_1 causally precedes e_2 in the VT protocol, i.e. $VT_i = \{\langle e_1, e_2 \rangle \mid e_1.VT \leq e_2.VT\}$.

Here, $TV_i - TV\_VT_i$, $TV\_LT_i - TV\_VT_i$, $TV\_PT_i - TV\_VT_i$, and $HTBC_i - TV\_VT_i$ are sets of pairs of event messages which are unnecessarily ordered by a peer p_i in the TV, TV_ LT, TV_PT, and HTBC protocols, respectively. For example, $HTBC_i - TV\_VT_i$ is a set $\{\langle e_1, e_2 \rangle \mid e_1 \Rightarrow_{TV} e_2$, $e_1 \Rightarrow_{LT} e_2$ and $e_1 \Rightarrow_{PT} e_2$, but

$e_1.TV \not\leq e_2.TV$ and $e_1.VT \not\leq e_2.VT\}$ of pairs of event messages. Here, $e_1 \Rightarrow_{HTBC} e_2$ but e_1 does not causally precede e_2. For each set M of pairs of ordered event messages, $N(M)$ shows the unnecessarily ordered event message (UO) number. For example, the UO number $N(HTBC_i - TV_{VT_i})$ shows how many number of event messages unnecessarily precede each event message. In the TV_VT protocol, $N(TV\_VT_i) = \emptyset$ since $e_1 \Rightarrow_{TV\_VT} e_2$ if $e_1 \rightarrow e_2$. Hence, there is no pair of event messages unnecessarily ordered in the TV_VT protocol $N(TV\_VT_i) = \emptyset$.

Figure 1 shows the UO number $N(TV_i - TV\_VT_i)$, $N(TV\_LT_i - TV\_VT_i)$, $N(TV\_PT_i - TV\_VT_i)$, and $N(HTBC_i - TV\_VT_i)$ of every peer p_i for the maximum clock offset ε is 4 [tu] and number of subgroups is 2 where event messages are randomly transmitted by six peers (n = 6). As shown in Figure 1, the UO number of event messages in the HTBC protocol is a little smaller than the TV_PT protocol. The UO number of event messages can be reduced in the HTBC protocol.

4. CONCLUDING REMARKS

In this paper, we proposed the topic-based peer-to-peer publish/subscribe (P2PPS) model where each peer can play both

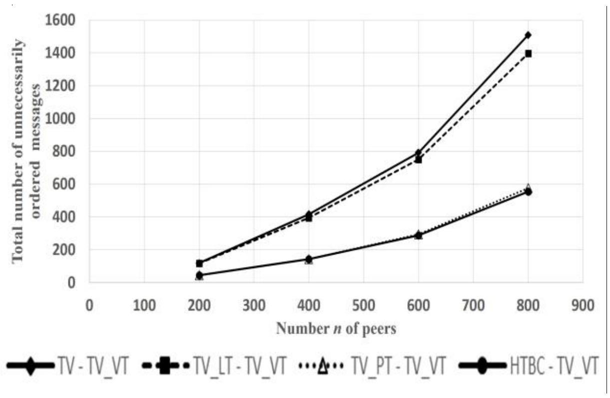

Figure 1. Total number of unnecessarily ordered event messages in heterogeneous network (n = 6, *maxd* = 16 in LAN, *maxd* = 160 in WAN).

subscriber and publisher roles and there is no centralized coordinator. Each peer can not only subscribe but also publish event messages. Event messages published by a peer are delivered to only target peers where the subscriptions and publications include common topics. We discussed the HTBC, TV_LT, TV_PT, and TV protocols which take advantage of topic vectors, linear time, and physical time to causally deliver event messages to every common target peer. Even if an event message e_1 is delivered before e_2 in a protocol, e_1 may not causally precede e_2. Here, the event messages e_1 and e_2 are unnecessarily ordered in the protocol. We evaluated the HTBC, TV_LT, TV_ PT, and TV protocols in terms of number of pairs of unnecessarily ordered event messages. In the evaluation, we showed there are only a small number of event messages are unnecessarily preceded by each event message in the HTBC protocol.

ACKNOWLEDGMENT

We would like to thank Prof. Makoto Takizawa for supervising to our thesis.

REFERENCES

1) L. Lamport, "Time, Clocks, and the Ordering of Events in a Distributed System," *Communications of the ACM*, vol.21, no.7, 1978, pp.558–565.
2) F. Mattern, "Virtual Time and Global States of Distributed Systems," *North-Holland: Amsterdam*, 1989, pp.215–226.
3) H. Nakayama, D. Duolikun, T. Enokido, and M. Takizawa, "Synchronization of Peers in Peer-to-peer Publish/Subscribe Systems," *Proc. of the 9th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS-2015)*, 2015, pp.252-259.
4) H. Nakayama, D. Duolikun, T. Enokido, and M. Takizawa, "Selective Delivery of Event Messages in Peer-to-peer Topic-based Publish/Subscribe Systems," *Proc. of the 18th International Conference on Network-Based Information Systems (NBiS-2015)*, 2015, pp.379-386.
5) H. Nakayama, D. Duolikun, T. Enokido, and M. Takizawa, "Reduction of Unnecessarily Ordered Event Messages in Peer-to-peer Model of Topic-based Publish/Subscribe Systems," *Proc. of IEEE the 30th International Conference on Advanced Information Networking and Applications (AINA-2016)*, 2016, pp.1160-1167.
6) H. Nakayama, S. Nakamura, T. Enokido, and M. Takizawa, "Scalable Group Communication Protocols in the Peer-to-peer Model of Topic-based Publish/Subscribe Systems," *Proc. of the International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS-2016)*, 2016, pp.142-149.
7) V. Setty, M. V. Steen, R. Vitenberg, and S. Voulgaris, "PolderCast: Fast, Robust, and Scalable Architecture for P2P Topic-based Pub/Sub," *Proc. of ACM/IFIP/USENIX 13th International Conference on Middleware (Middleware 2012)*, pp.271-291, 2012.
8) S. Tarkoma, M. Ain and K. Visala, "The Publish/Subscribe Internet Routing Paradigm (PSIRP) : Designing the Future Internet Architecture," *Future Internet Assembly*, pp.102-111, 2009.
9) S. Tarkoma, "Publish/Subscribe System : Design and Principles (First Edition)," *John Wiley and Sons, Ltd*, 2012, 360pages.

10) A. B. Waluyo, D. Taniar, W. Rahayu, A. Aikebaier, M. Takizawa, and B. Srinivasan, "Trustworthy-based Efficient Data Broadcast Model for P2P Interaction in Resource-Constrained Wireless Environments," *Journal of Computer and System Sciences (JCSS)*, Vol.78, No.6, pp.1716-1736, 2012.
11) K. Watanabe, Y. Nakajima, T. Enokido, and M. Takizawa, "Ranking Factors in Peer-to-Peer Overlay Networks," *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, Vol.2, No.3, Article 11, pp.11:1-11:26, 2007.