Timely Hybrid Synchronous Virtual Networks

Authors

Rasha Ghassan Hasan
Fernando Luis Dotti

Pontificia Universidade Católica do Rio Grande do Sul (PUCRS)
Porto Alegre - Brazil
Motivation: Hybrid synchronous models

Hybrid Synchrony Virtual Networks

Performance Evaluation

Related Works
Distributed algorithms take into account the synchrony level provided by the system.

Asynchronous systems:
- Time bounds do not exist: The processing time and network delay may vary greatly.

Synchronous systems:
- Time bounds exist: processing time and network delay are bounded.

Hybrid Synchronous model:
- Assume intermediate levels between synchrony and asynchrony [1,2].

Distributed Systems hybrid in space example: failure detector

Perfect failure detector on hybrid synchrony environment [3]

Distributed Systems hybrid in **time** example: consensus problem

Consensus algorithm proposed in [5], first round with P1 coordinator

---

The timed-asynchronous model assumes that the system alternate between synchronous and asynchronous behavior [5].

For each execution, there is a time after which the upper bound $\delta$ is respected by the system. This time is called Global Stabilization Time (GST).

Since the upper bound cannot hold forever, it is accepted that it holds just for a limited time $\Delta_s$.

In practical terms, $\Delta_s$ is the time needed for consensus to make progress or to be reached.

---

Many applications benefit from consensus and failure detection as building blocks in the distributed algorithms, for example Apache Cassandra [6]

Motivation: Hybrid synchronous models

Hybrid Synchrony Virtual Networks

Performance Evaluation

Related Works
**Hybrid Synchronous Virtual Networks (HSVNs)**

- **Virtual Networks (VN) [7]**
  - Share resources → cost
  - Tasks distinguishments
  - Resilience
  - Resources allocation
  - ...

- **Distributed Systems (DS) with Hybrid Synchrony [1,2]**

---

**Hybrid Synchrony Virtual Networks**

Work history

1. Space_HSVN [8,9,10]

2. Time_HSVN (current work)

1. Define the assumptions and abstractions needed to characterize both **Substrate Networks (SNs)** and **Virtual Networks (VNs)** suitable for Time HSVNs

2. Develop an embedding model for Time HSVNs that
   (i) answers the timely synchronous nature of the system
   (ii) aware of sparing synchronous resources which are relatively expensive.
1. Time-HSVN characterization

➢ Reflect the nature of timely synchronous distributed systems, which repeatedly demand eventual synchrony during the system life,
➢ VNs with a cyclic pattern - cycle T time units,
➢ During T, each virtual node and link demands synchrony once, for a certain period,
➢ The client needs to be provided the required synchrony eventually within T.
The HSVNs demand synchrony once during $T \rightarrow 1$ synchronous frame / $T$,

The synchronous frame is partitioned into synchronous slots,

The virtual demands mapped to a synchronous slot should not violate the physical BW capacity to eliminate competition and assure synchrony,

---

Objective: How to map the virtual synchronous slots to the physical synchronous slots, with the objective of minimizing the mapping cost represented by the used BW?

Approach: 1. refine the model of Space HSVNs mapping [8,9,10]
    2. enhanced the achieved solution to allow more VNs to be mapped.

Timely Hybrid Synchronous Virtual Networks

3. Embedding phases

Time - HSVN mapping block diagram inspired from [11]

SN: Substrate Network
VNs: Virtual Networks
BW: Band Width
snc.: synchrony

Motivation: Hybrid synchronous models

Hybrid Synchrony Virtual Networks

Performance Evaluation

Related Works
Work load and tools

- MIP implemented in ZIMPL language,
- BRITE tool for generating topologies,
- CPLEX optimizer.

<table>
<thead>
<tr>
<th>Expe.</th>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>A4</th>
<th>B1</th>
<th>B2</th>
<th>B3</th>
<th>B4</th>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
</tr>
</thead>
<tbody>
<tr>
<td>VN size</td>
<td>10 nodes</td>
<td></td>
<td></td>
<td></td>
<td>20 nodes</td>
<td></td>
<td></td>
<td></td>
<td>30 nodes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Virtual links sync. slots</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Virtual nodes sync. each VN size</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 slot per ( T )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VN sizes</td>
<td>3,4,5 nodes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VNs BW</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>uniformly distributed: 100Mbps-1Gbps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VNs CPU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10,15,25 % of SN nodes CPU</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SN size</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>15 nodes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SN BW</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>uniformly distributed: 1 Gbps-3 Gbps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SN CPU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>nodes fully free initially</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Experiments parameters

MIP: Mixed Integer Program
ZIMPL: Zuse Institute Mathematical Programming Language
BRITE: Boston university Representative Internet Topology gEnerator
➢ within each experiment’s group, the BW used with the space model is equal to the maximum BW used within the group, and the spared ratio increases when the synchronous demands within T decreases.
The Substrate Network resources seem to have load distribution
most of the scenarios demanded optimization time that is less than 20 minutes,

The time model demands more time than the space model. And we notice that the difference between both increases with the increment of the problem size.
Topological study

➢ The model tends towards reserving more physical elements with the increment of the synchronous slots demanded by the VNs,
➢ the model tends also towards distributing the synchrony load.

Topology divergence of used physical resource

(a) scenario A1  (b) scenario A2  (c) scenario A3
Micro-phase efficiency

The efficiency decreases when the virtual links load increases,

- the efficiency increases when the maximum number of synchronous slots demanded decreases,

- the micro model efficiency is the same in group N and O: both groups are with high virtual links load → NO slots sharing → NO optimization for the macro-map solution.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>VN load/SN capacity</th>
<th>$sync(i^k, j^k)=1,2,3$ slots</th>
<th>$sync(i^k, j^k)=1$ slot</th>
</tr>
</thead>
<tbody>
<tr>
<td>K</td>
<td>(0-20] %</td>
<td>9</td>
<td>13</td>
</tr>
<tr>
<td>L</td>
<td>(20-40] %</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>M</td>
<td>(40-60] %</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>N</td>
<td>(60-80] %</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>O</td>
<td>(80-100] %</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

Number of mapped virtual links with different load and synchrony demands
Motivation: Hybrid synchronous models
Hybrid Synchrony Virtual Networks
Performance Evaluation
Related Works
Related Works


Timely Hybrid Synchrony Virtual Networks

- **What?**

Virtual Networks with subsets of nodes and links that eventually obey time bounds for processing and communication

- **Why?**

Hosting distributed systems with hybrid synchrony in time with while economizing the use of the synchronous resources

- **How?**

→ A substrate network with hybrid synchronous resources in space
→ embedding model that uses the synchronous resources attentively
Timely Hybrid Synchronous Virtual Networks

Authors

Rasha Ghassan Hasan

Fernando Luis Dotti

rasha.hasan@acad.pucrs.br
fernando.dotti@pucrs.br