Analyzing Paxos with Fault-Tolerant Multiparty Session Types

Bachelor thesis by Nicolas Daniel Torres
Date of submission: November 8, 2021

1. Review: Prof. Dr. Kirstin Peters
Darmstadt

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Computer Science
Department
<institute>
<working group>

Contents

1 Introduction

In distributed systems components on different computers coordinate and communicate via message passing to achieve a common goal. Sometimes, to achieve this goal, the individual components need to reach consensus, i.e., agree on the value of some data using a consensus algorithm. For example in state machine replication or when deciding which database transactions should be committed in what order. For such a distributed system to behave correctly the consensus algorithm needs to be correct. Thus, analyzing consensus algorithms is important.

To achieve consensus, consensus algorithms must satisfy the following properties: termination, validity, and agreement [1]. Proving these properties can be complicated. Model checking tools lead to big state-spaces so static analysis is preferable. For static analysis Multiparty Session Types are particularly interesting because session typing can ensure protocol conformance and the absence of communication errors and deadlocks [4].

Due to the presence of faulty processes and unreliable communication consensus algorithms are designed to be fault-tolerant. Modelling fault-tolerance is not possible using Multiparty Session Types, thus a fault-tolerant extension is necessary. Peters, Nestmann, and Wagner developed such an extension called Fault-Tolerant Multiparty Session Types.

In this work we will use Fault-Tolerant Multiparty Session Types to analyze the consensus algorithm Paxos, as described in [2].

2 Technical Preliminaries

First, we define the sorts, some additional notation, and use them to define the global type. Afterwards we define some sets and functions to create the processes.

3 Model and Analysis

First, we specify some sorts with which we can then define the global type. Afterwards, we define the processes for the proposer and the acceptor. Finally, we will study an example run of the model.

3.1 Sorts

We assume the following sorts.

$\text{Maybe } a = \{\text{Just } a, \text{Nothing}\}$

$\text{Bool} = \{\text{true}, \text{false}\}$

$\text{Value} = $ Set of values.

$\text{Promise } a = \{\text{Promise } (\text{Maybe } (\text{Proposal } a)), \text{Nack } \mathbb{N}\}$

$\text{Proposal } a = \{\text{Proposal } \mathbb{N}\, a\}$

3.2 Global Type

Since each proposer initiates its own session the global type can be defined for one proposer. A quorum of acceptors A_Q is assumed.

The last phase of Paxos contains no inter-process communication, so it is not modeled in the global type.

$$G_{p,A_Q} = (\mu X) \bigodot_{a \in A_Q} p \to_u a : l1a \langle \mathbb{N} \rangle . \bigodot_{a \in A_Q} a \to_u p : l1b \langle \text{Promise Value} \rangle .$$
$$p \to_w A_Q : Accept. \left(\bigodot_{a \in A_Q} p \to_u a : l2a \langle \text{Proposal Value} \rangle \right) .end$$

$\oplus\ Restart.X$
$\oplus\ Abort.end$

We can distinguish the individual phases of the Paxos algorithm by the labels $l1a$, $l1b$, and $l2a$.

In the first two steps, $1a$ and $1b$, the proposer sends its proposal number to each acceptor in A_Q and listens for their responses. In step 2a the proposer decides whether to send an $Accept$ or $Restart$ message to restart the algorithm. This decision is broadcast to all acceptors in A_Q. Should the proposer crash the algorithm ends for this particular proposer and quorum of acceptors.

3.3 Functions

We define some functions which we use in the next section to define the processes.

$\mathrm{proposalNumber} : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$

$\mathrm{proposalNumber}\,(a, b)$ returns a proposal number when given two natural numbers.

$\mathrm{promiseValue} : [\mathrm{Promise}\ \ a] \to a$

$\mathrm{promiseValue}\,(ps)$ returns a new value if none of the promises in ps contain a value. Otherwise, the best value is returned. Usually, that means the value with the highest associated proposal number. A promise contains a value v if it is of the form $\mathrm{Promise}\ (\mathrm{Just}\ \ v)$.

$\mathrm{anyNack} : [\mathrm{Promise}\ \ a] \to \mathrm{Bool}$
$\mathrm{anyNack}\,([]) = false$
$\mathrm{anyNack}\,((\mathrm{Nack}\ \ \_ : \_)) = true$
$\mathrm{anyNack}\,((\_ : xs)) = \mathrm{anyNack}\,(xs)$

$\mathrm{anyNack}\,(ps)$ returns true if the list contains at least one promise of the form $\mathrm{Nack}\ \ n$. Otherwise, it returns false.

$\mathrm{promiseCount} : [\mathrm{Promise}\ \ a] \to \mathbb{N}$
$\mathrm{promiseCount}\,([]) = 0$
$\mathrm{promiseCount}\,((\mathrm{Promise}\ \ \_ : xs)) = 1 + \mathrm{promiseCount}\,(xs)$
$\mathrm{promiseCount}\,((\_ : xs)) = \mathrm{promiseCount}\,(xs)$

$\mathrm{promiseCount}\,(ps)$ takes a list of promises ps and calculates the number of promises in that list of that have the form $\mathrm{Promise}\ \ m$.

$\mathrm{gt} : a \to \mathrm{Maybe}\ \ a \to \mathrm{Bool}$
$\mathrm{gt}\,(\_, \mathrm{Nothing}) = \mathit{true}$
$\mathrm{gt}\,(a, \mathrm{Just}\ \ b) = a > b$

$\mathrm{ge} : a \to \mathrm{Maybe}\ \ a \to \mathrm{Bool}$
$\mathrm{ge}\,(\_, \mathrm{Nothing}) = \mathit{true}$
$\mathrm{ge}\,(a, \mathrm{Just}\ \ b) = a \geq b$

$\mathrm{nFromProposal} : \mathrm{Proposal}\ \ a \to \mathbb{N}$
$\mathrm{nFromProposal}\,(\mathrm{Proposal}\ \ n\ \_) = n$

$\mathrm{nFromProposal}\,(p)$ retrieves the proposal number n inside proposal p, which has the form $\mathrm{Proposal}\ \ n\ pr$.

$\mathrm{genA_Q} : \mathbb{N} \times \mathbb{N} \times \mathbb{N} \to \mathbb{N}$

$\mathrm{genA_Q}\,(i, ac, pc)$ returns a randomly selected set A_Q with $A_Q \subseteq A = \{1, \ldots, ac\}$ and $|A_Q| > \frac{|A|}{2}$.

$\mathrm{update}\,(n, m)$ replaces the value inside n with the value in m.

3.4 Processes

3.4.1 System Initialization

$\mathrm{Sys}\,(ac, pc) = \overline{a}\,[2]\,(t)\,.\,\mathrm{P}^{\mathrm{p}}_{\mathrm{init}}\,(ac + 1, \mathrm{genA_Q}\,(ac + pc, ac, pc)\,, ac + pc, ac + pc, [])$
$|\ a\,[1]\,(t)\,.\Pi_{ac<i<ac+pc}\ \mathrm{P}^{\mathrm{p}}_{\mathrm{init}}\,(ac + 1, \mathrm{genA_Q}\,(i, ac, pc)\,, i, i, [])$
$|\ \Pi_{1\leq j\leq ac}\ \mathrm{P}^{\mathrm{a}}_{\mathrm{init}}\,(j, ac + 1, ac, pc, n_j, pr_j)$

$$\mathrm{P}^{\mathrm{p}}_{\mathrm{init}}\left(i, A_Q, n, m, \overrightarrow{V}\right) = \overline{b_n}\,[i]\,(s)\,.\,\mathrm{P}^{\mathrm{p}}$$

$$\mathrm{P}^{\mathrm{a}}_{\mathrm{init}}\,(j, i, ac, pc, n, pr) = \Pi_{ac<k\leq ac+pc}\ b_k\,[j]\,(s)\,.\,\mathrm{P}^{\mathrm{a}}$$

$\mathrm{Sys}\,(ac, pc)$, $\mathrm{P}^{\mathrm{p}}_{\mathrm{init}}\left(i, A_Q, n, m, \overrightarrow{V}\right)$, and $\mathrm{P}^{\mathrm{a}}_{\mathrm{init}}\,(j, i, ac, pc, n, pr)$ describe the system initialization. ac and pc are the number of acceptors and proposers respectively.

An outer session is created through shared-point a. This outer session is not strictly necessary but was left in to allow for easier extension of the model. The acceptors are initialized using indices from 1 to ac and the proposers are initialized using indices from $ac + 1$ to $ac + pc$.

$\mathrm{P}^{\mathrm{p}}_{\mathrm{init}}\left(i, A_Q, n, m, \overrightarrow{V}\right)$ is initialized with the proposer's role in its own session i, which is always $ac+1$, a quorum of acceptors A_Q, an index n, and a vector $\overrightarrow{V}$. Each proposer has the same role $i = ac+1$ but uses a different shared-point b_n according to its index n. $\overrightarrow{V}$ is used in the proposer to collect and evaluate the responses from the acceptors. It is always initialized with an empty list []. Shared-point b_n is used to initiate a session. Afterwards, the process behaves like P^{p}.

$\mathrm{P}^{\mathrm{a}}_{\mathrm{init}}\left(j, i, ac, pc, n, pr\right)$ is initialized with the acceptor's index j, the proposer index i, which is always $ac+1$, ac, pc, initial knowledge for the highest promised proposal number n, if available, and initial knowledge for the most recently accepted proposal pr, if available. n is of type $\mathrm{Maybe}\ \mathbb{N}$ and pr is of type $\mathrm{Maybe}\ (\mathrm{Proposal}\ \ \mathrm{Value})$ thus both can be $\mathrm{Nothing}$. Each of the proposers' session requests are accepted in a separate subprocess. These subprocesses run parallel to each other but still access the same values for n and pr. We observe that each subprocess in an acceptor accesses a different channel s, since it is generated by the proposer and passed through when the proposers' session request is accepted. Afterwards, each subprocess behaves like P^{a}.

3.4.2 Proposer

$$
\begin{aligned}
&\mathrm{P}^{\mathrm{p}} = (\mu X)\, \mathrm{update}\,(n, n+1)\,.\\
&\quad \left(\textstyle\bigodot_{j\in A_Q}\ s\,[i,j]!_u l1a\,\langle \mathrm{proposalNumber}\,(n,m)\rangle\right).\\
&\quad \left(\textstyle\bigodot_{j\in A_Q}\ s\,[j,i]?_u l1b\,\langle \bot\rangle\,(v_j)\right).\\
&\quad \text{if}\ \ \mathrm{anyNack}\left(\overrightarrow{V}\right)\ \text{ or }\ \mathrm{promiseCount}\left(\overrightarrow{V}\right) < \left\lceil \tfrac{i}{2}\right\rceil\\
&\quad\ \ \text{then}\ \ s\,[i, A_Q]!_w \mathit{Restart}.X\\
&\quad\ \ \text{else}\\
&\quad\ \ \ s\,[i, A_Q]!_w \mathit{Accept}.\\
&\quad\ \ \ \textstyle\bigodot_{j\in A_Q}\ s\,[i,j]!_u l2a\left\langle \mathrm{Proposal}\ \ \mathrm{proposalNumber}\,(n,m)\ \ \mathrm{promiseValue}\left(\overrightarrow{V}\right)\right\rangle.\\
&\quad\ \ \ \mathit{end}
\end{aligned}
$$

At the start of the recursion n is incremented to make sure every run of the recursion uses a different n and thus a different proposal number. The proposal number is sent to every acceptor in A_Q and their replies are gathered in $\overrightarrow{V}$ through v_j. Because $i = ac+1$, the minimum number of acceptors needed to form a majority is $\left\lceil \frac{i}{2}\right\rceil = \left\lceil \frac{ac+1}{2}\right\rceil$. If any $\mathrm{Nack}\ \ x$ was received or the number of $\mathrm{Promise}\ \ y$ received is less than that needed for the smallest majority the proposer restarts the algorithm. Otherwise, the proposer sends its proposal to the acceptors and terminates.

3.4.3 Acceptor

```
$\mathrm{P^a} = (\mu X)\, s\,[i,j]?_u l1a\,\langle\bot\rangle\,(n')\,.$
 if $n' =\bot$
  then $\mathrm{P^a_{cont}}$
  else
   if $\mathrm{gt}\,(n', n)$
   then $\mathrm{update}\,(n, n')\,.s\,[j,i]!_u l1b\,\langle\mathrm{Promise}\ pr\rangle\,.\,\mathrm{P^a_{cont}}$
   else $s\,[j,i]!_u l1b\,\langle\mathrm{Nack}\ n\rangle\,.\,\mathrm{P^a_{cont}}$

$\mathrm{P^a_{cont}} = s\,[i,j]?_w Accept.s\,[i,j]?_u l2a\,\langle\bot\rangle\,(pr')\,.$
  if $pr' =\bot$
   then $end$
   else
    if $\mathrm{ge}\,(\mathrm{nFromProposal}\,(pr')\,, n)$
     then $\mathrm{update}\,(pr, pr')\,.\,\mathrm{update}\,(n, \mathrm{Just}\ \ \mathrm{nFromProposal}\,(pr'))\,.end$
     else $end$
 $\oplus\ Restart.X$
 $\oplus\ Abort.end$
```

For each proposer an acceptor has a corresponding subprocess, which behaves like $\mathrm{P^a}$. These subprocesses access the same values for n and pr. This means that updating these values with $\mathrm{update}\,(n, m)$ updates them for all subprocesses of an acceptor.

Each subprocess can communicate with one proposer. Thus, if that proposer does not or can not communicate with a particular subprocess of an acceptor then there is no need for that subprocess. It is possible that an acceptor participates in a proposers' session but is not contained in the proposers' quorum of acceptors A_Q, in which case the proposer does not communicate with that acceptor. It is also possible for a proposer to crash or otherwise terminate, in which case the proposer can not communicate with that acceptor.

Each subprocess starts out by potentially receiving a proposal number n' from the corresponding proposer. If the acceptor does receive a proposal number n' it responds with either $\mathrm{Promise}\ \ pr$ or $\mathrm{Nack}\ \ n$, depending on the values of n' and n. If the acceptor does not receive a proposal number then it sends no response to the proposer. In either case the subprocess moves on to receive the proposers' decision in phase $2a$.

Since the proposers' decision broadcast is weakly reliable, there are two cases in which the acceptor receives no decision. The proposer might have terminated or this particular acceptor is not in the proposers' quorum of acceptors A_Q. In either case this particular subprocess of the acceptor is no longer needed, because each subprocess of the acceptor exclusively communicates with one proposer. Thus, the subprocess terminates in the default branch $Abort$.

In the $Restart$ branch this particular subprocess of the acceptor restarts the algorithm to match the corresponding proposer.

In the $Accept$ branch the acceptor potentially receives a proposal pr' from the corresponding proposer. The acceptor updates n and pr if the proposal number in pr' is greater or equal to n. Then the subprocess terminates. If the acceptor does not receive a proposal or the proposal number of pr' is less than n the subprocess terminates without updating n or pr.

3.5 Failure Patterns

In Paxos there is no need to reject outdated messages so $\mathrm{FP}_{\mathrm{uget}}$ is implemented with a constant true.

$\mathrm{FP}_{\mathrm{uskip}}$

$\mathrm{FP}_{\mathrm{wskip}}$

$\mathrm{FP}_{\mathrm{ml}}$

$\mathrm{FP}_{\mathrm{crash}}$

3.6 Example

$ac = 3 \; pc = 2 \; V = \{\mathrm{abc}\}$

$$\begin{aligned}
&\mathrm{Sys}\,(3,2) = a\,[1]\,(t)\,.\Pi_{3<i<5}\ \mathrm{P}^{\mathrm{p}}_{\mathrm{init}}\,(4, \mathrm{genA_Q}\,(i,3,2)\,, i, i, [])\\
&\mid \overline{a}\,[2]\,(t)\,.\,\mathrm{P}^{\mathrm{p}}_{\mathrm{init}}\,(4, \mathrm{genA_Q}\,(5,3,2)\,, 5, 5, [])\\
&\mid \Pi_{1\leq j\leq 3}\ \mathrm{P}^{\mathrm{a}}_{\mathrm{init}}\,(j, 4, 3, 2, n_j, pr_j)\\
&\longmapsto (\mathrm{Init})
\end{aligned}$$

$$
\begin{aligned}
&(\nu t)\, \big(\mathrm{P}^{\mathrm{p}}_{\text{init}}\,(4, A_{Q,1}, 4, 4, [])\ \mid\ \mathrm{P}^{\mathrm{p}}_{\text{init}}\,(4, A_{Q,2}, 5, 5, [])\\
&\mid\ \mathrm{P}^{\mathrm{a}}_{\text{init}}\,(1, 4, 3, 2, n_1, pr_1)\ \mid\ \mathrm{P}^{\mathrm{a}}_{\text{init}}\,(2, 4, 3, 2, n_2, pr_2)\ \mid\ \mathrm{P}^{\mathrm{a}}_{\text{init}}\,(3, 4, 3, 2, n_3, pr_3)\\
&\mid \Pi_{1\le k,l\le 2,k\neq l}\ t_{k\to l} : []\big)
\end{aligned}
$$

$$
\begin{aligned}
&=\\
&(\nu t)\, \big(\overline{b_4}\,[4]\,(s)\,.\,\mathrm{P}^{\mathrm{p}}\ \mid \overline{b_5}\,[4]\,(r)\,.\,\mathrm{P}^{\mathrm{p}}\\
&\mid \Pi_{3<n\le 5}\ b_n\,[1]\,(s)\,.\,\mathrm{P}^{\mathrm{a}}\\
&\mid \Pi_{3<n\le 5}\ b_n\,[2]\,(s)\,.\,\mathrm{P}^{\mathrm{a}}\\
&\mid \Pi_{3<n\le 5}\ b_n\,[3]\,(s)\,.\,\mathrm{P}^{\mathrm{a}}\\
&\mid \Pi_{1\le k,l\le 2,k\neq l}\ t_{k\to l} : []\big)\\
&=\\
&(\nu t)\, \big(\overline{b_4}\,[4]\,(s)\,.\,\mathrm{P}^{\mathrm{p}}\ \mid \overline{b_5}\,[4]\,(r)\,.\,\mathrm{P}^{\mathrm{p}}\\
&\mid\ (b_4\,[1]\,(s)\,.\,\mathrm{P}^{\mathrm{a}}\ \mid b_5\,[1]\,(r)\,.\,\mathrm{P}^{\mathrm{a}})\\
&\mid\ (b_4\,[2]\,(s)\,.\,\mathrm{P}^{\mathrm{a}}\ \mid b_5\,[2]\,(r)\,.\,\mathrm{P}^{\mathrm{a}})\\
&\mid\ (b_4\,[3]\,(s)\,.\,\mathrm{P}^{\mathrm{a}}\ \mid b_5\,[3]\,(r)\,.\,\mathrm{P}^{\mathrm{a}})\\
&\mid \Pi_{1\le k,l\le 2,k\neq l}\ t_{k\to l} : []\big)
\end{aligned}
$$

$$
\begin{aligned}
&\longmapsto (\mathit{Init})\\
&(\nu t)\,(\nu s)\,(\nu r)\,\big((\mu X)\,\text{update}\,(n,5)\,.\,\Big(\bigodot_{j\in\{1,2\}}\ s\,[4,j]!_u l1a\,\langle\text{proposalNumber}\,(n,4)\rangle\Big)\ldots\\
&\mid\ (\mu X)\,\text{update}\,(n,6)\,.\,\Big(\bigodot_{j\in\{2,3\}}\ r\,[4,j]!_u l1a\,\langle\text{proposalNumber}\,(n,5)\rangle\Big)\ldots\\
&\mid\ ((\mu X)\,s\,[4,1]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots\ \mid\ (\mu X)\,r\,[4,1]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots)\\
&\mid\ ((\mu X)\,s\,[4,2]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots\ \mid\ (\mu X)\,r\,[4,2]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots)\\
&\mid\ ((\mu X)\,s\,[4,3]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots\ \mid\ (\mu X)\,r\,[4,3]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots)\\
&\mid \Pi_{1\le k,l\le 4,k\neq l}\ s_{k\to l} : []\ \mid \Pi_{1\le k,l\le 4,k\neq l}\ r_{k\to l} : []\ \mid \Pi_{1\le k,l\le 2,k\neq l}\ t_{k\to l} : []\big)
\end{aligned}
$$

$$
\begin{aligned}
&=\\
&(\nu t)\,(\nu s)\,(\nu r)\,\big((\mu X)\,s\,[4,1]!_u l1a\,\langle\text{proposalNumber}\,(5,4)\rangle\,.s\,[4,2]!_u l1a\,\langle\text{proposalNumber}\,(5,4)\rangle\ldots\\
&\mid\ (\mu X)\,r\,[4,2]!_u l1a\,\langle\text{proposalNumber}\,(6,5)\rangle\,.r\,[4,3]!_u l1a\,\langle\text{proposalNumber}\,(6,5)\rangle\ldots\\
&\mid\ ((\mu X)\,s\,[4,1]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots\ \mid\ (\mu X)\,r\,[4,1]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots)\\
&\mid\ ((\mu X)\,s\,[4,2]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots\ \mid\ (\mu X)\,r\,[4,2]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots)\\
&\mid\ ((\mu X)\,s\,[4,3]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots\ \mid\ (\mu X)\,r\,[4,3]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots)\\
&\mid \Pi_{1\le k,l\le 4,k\neq l}\ s_{k\to l} : []\ \mid \Pi_{1\le k,l\le 4,k\neq l}\ r_{k\to l} : []\ \mid \Pi_{1\le k,l\le 2,k\neq l}\ t_{k\to l} : []\big)
\end{aligned}
$$

$$
\begin{aligned}
&\longmapsto (\mathit{USend} + \mathit{UGet})\\
&(\nu t)\,(\nu s)\,(\nu r)\,\big((\mu X)\,s\,[4,1]!_u l1a\,\langle\text{proposalNumber}\,(5,4)\rangle\,.s\,[4,2]!_u l1a\,\langle\text{proposalNumber}\,(5,4)\rangle\ldots\\
&\mid\ (\mu X)\,r\,[4,3]!_u l1a\,\langle\text{proposalNumber}\,(6,5)\rangle\,.r\,[2,4]?_u l1b\,\langle\perp\rangle\,(v_2)\ldots\\
&\mid\ ((\mu X)\,s\,[4,1]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots\ \mid\ (\mu X)\,r\,[4,1]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots)\\
&\mid\ ((\mu X)\,s\,[4,2]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots\ \mid\ (\mu X)\,\text{if}\ \ 10 =\perp\ \text{then}\ \mathrm{P}^{\mathrm{a}}_{\text{cont}}\ \text{else}\ \ldots)
\end{aligned}
$$

$| \; ((\mu X)\, s\,[4,3]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots \;|\; (\mu X)\, r\,[4,3]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots)$
$| \; \Pi_{1\le k,l\le 4, k\neq l}\; s_{k\to l} : [] \;|\; \Pi_{1\le k,l\le 4, k\neq l}\; r_{k\to l} : [] \;|\; \Pi_{1\le k,l\le 2, k\neq l}\; t_{k\to l} : [])$

$=$

$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[4,1]!_u l1a\,\langle \text{proposalNumber}\,(5,4)\rangle\,.s\,[4,2]!_u l1a\,\langle \text{proposalNumber}\,(5,4)\rangle\ldots$
$| \; (\mu X)\, r\,[4,3]!_u l1a\,\langle \text{proposalNumber}\,(6,5)\rangle\,.r\,[2,4]?_u l1b\,\langle\bot\rangle\,(v_2)\ldots$
$| \; ((\mu X)\, s\,[4,1]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots \;|\; (\mu X)\, r\,[4,1]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots)$
$| \; ((\mu X)\, s\,[4,2]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots \;|\; (\mu X)\,\text{if}\;\;\text{gt}\,(10,\text{Nothing})\,\text{then}\;\;\text{update}\,(n,n')\ldots\text{else}\;\ldots)$
$| \; ((\mu X)\, s\,[4,3]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots \;|\; (\mu X)\, r\,[4,3]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots)$
$| \; \Pi_{1\le k,l\le 4, k\neq l}\; s_{k\to l} : [] \;|\; \Pi_{1\le k,l\le 4, k\neq l}\; r_{k\to l} : [] \;|\; \Pi_{1\le k,l\le 2, k\neq l}\; t_{k\to l} : [])$

$=$

$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[4,1]!_u l1a\,\langle \text{proposalNumber}\,(5,4)\rangle\,.s\,[4,2]!_u l1a\,\langle \text{proposalNumber}\,(5,4)\rangle\ldots$
$| \; (\mu X)\, r\,[4,3]!_u l1a\,\langle \text{proposalNumber}\,(6,5)\rangle\,.r\,[2,4]?_u l1b\,\langle\bot\rangle\,(v_2)\ldots$
$| \; ((\mu X)\, s\,[4,1]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots \;|\; (\mu X)\, r\,[4,1]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots)$
$| \; ((\mu X)\, s\,[4,2]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots \;|\; (\mu X)\, r\,[2,4]!_u l1b\,\langle \text{Promise}\;\;\text{Nothing}\rangle\,.\,\text{P}^{\text{a}}_{\text{cont}})$
$| \; ((\mu X)\, s\,[4,3]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots \;|\; (\mu X)\, r\,[4,3]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots)$
$| \; \Pi_{1\le k,l\le 4, k\neq l}\; s_{k\to l} : [] \;|\; \Pi_{1\le k,l\le 4, k\neq l}\; r_{k\to l} : [] \;|\; \Pi_{1\le k,l\le 2, k\neq l}\; t_{k\to l} : [])$

$\longmapsto (USend + UGet)$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[4,1]!_u l1a\,\langle \text{proposalNumber}\,(5,4)\rangle\,.s\,[4,2]!_u l1a\,\langle \text{proposalNumber}\,(5,4)\rangle\ldots$
$| \; (\mu X)\, r\,[2,4]?_u l1b\,\langle\bot\rangle\,(v_2)\,.r\,[3,4]?_u l1b\,\langle\bot\rangle\,(v_3)\ldots$
$| \; ((\mu X)\, s\,[4,1]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots \;|\; (\mu X)\, r\,[4,1]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots)$
$| \; ((\mu X)\, s\,[4,2]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots \;|\; (\mu X)\, r\,[2,4]!_u l1b\,\langle \text{Promise}\;\;\text{Nothing}\rangle\,.\,\text{P}^{\text{a}}_{\text{cont}})$
$| \; ((\mu X)\, s\,[4,3]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots \;|\; (\mu X)\, r\,[3,4]!_u l1b\,\langle \text{Promise}\;\;\text{Nothing}\rangle\,.\,\text{P}^{\text{a}}_{\text{cont}})$
$| \; \Pi_{1\le k,l\le 4, k\neq l}\; s_{k\to l} : [] \;|\; \Pi_{1\le k,l\le 4, k\neq l}\; r_{k\to l} : [] \;|\; \Pi_{1\le k,l\le 2, k\neq l}\; t_{k\to l} : [])$

$\longmapsto (USkip)$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[4,1]!_u l1a\,\langle \text{proposalNumber}\,(5,4)\rangle\,.s\,[4,2]!_u l1a\,\langle \text{proposalNumber}\,(5,4)\rangle\ldots$
$| \; (\mu X)\, r\,[2,4]?_u l1b\,\langle\bot\rangle\,(v_2)\,.r\,[3,4]?_u l1b\,\langle\bot\rangle\,(v_3)\ldots$
$| \; ((\mu X)\, s\,[4,1]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots \;|\; (\mu X)\,\text{if}\;\;\bot{=}\bot\;\;\text{then}\;\;\text{P}^{\text{a}}_{\text{cont}}\,\text{else}\;\;\ldots)$
$| \; ((\mu X)\, s\,[4,2]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots \;|\; (\mu X)\, r\,[2,4]!_u l1b\,\langle \text{Promise}\;\;\text{Nothing}\rangle\,.\,\text{P}^{\text{a}}_{\text{cont}})$
$| \; ((\mu X)\, s\,[4,3]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots \;|\; (\mu X)\, r\,[3,4]!_u l1b\,\langle \text{Promise}\;\;\text{Nothing}\rangle\,.\,\text{P}^{\text{a}}_{\text{cont}})$
$| \; \Pi_{1\le k,l\le 4, k\neq l}\; s_{k\to l} : [] \;|\; \Pi_{1\le k,l\le 4, k\neq l}\; r_{k\to l} : [] \;|\; \Pi_{1\le k,l\le 2, k\neq l}\; t_{k\to l} : [])$

$=$

$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[4,1]!_u l1a\,\langle \text{proposalNumber}\,(5,4)\rangle\,.s\,[4,2]!_u l1a\,\langle \text{proposalNumber}\,(5,4)\rangle\ldots$
$| \; (\mu X)\, r\,[2,4]?_u l1b\,\langle\bot\rangle\,(v_2)\,.r\,[3,4]?_u l1b\,\langle\bot\rangle\,(v_3)\ldots$
$| \; ((\mu X)\, s\,[4,1]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots \;|\; (\mu X)\, r\,[4,1]?_w Accept \cdots \oplus Restart.X \oplus Abort.end)$
$| \; ((\mu X)\, s\,[4,2]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if}\;\ldots \;|\; (\mu X)\, r\,[2,4]!_u l1b\,\langle \text{Promise}\;\;\text{Nothing}\rangle\,.\,\text{P}^{\text{a}}_{\text{cont}})$

$| \ ((\mu X)\, s\,[4,3]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if } \ldots \ | \ (\mu X)\, r\,[3,4]!_u l1b\,\langle\text{Promise Nothing}\rangle\,.\,\mathrm{P}^{\mathrm{a}}_{\mathrm{cont}})$
$| \ \Pi_{1\le k,l\le 4, k\ne l}\, s_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 4, k\ne l}\, r_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 2, k\ne l}\, t_{k\to l} : [])$

$\longmapsto (\mathit{USend} + \mathit{UGet})$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[4,1]!_u l1a\,\langle\text{proposalNumber}\,(5,4)\rangle\,.s\,[4,2]!_u l1a\,\langle\text{proposalNumber}\,(5,4)\rangle \ldots$
$| \ (\mu X)\, r\,[3,4]?_u l1b\,\langle\bot\rangle\,(v_3)\,.\,\text{if anyNack}\left(\overrightarrow{V}\right)\ \text{or promiseCount}\left(\overrightarrow{V}\right) < 1\,\text{then} \ \ldots$
$| \ ((\mu X)\, s\,[4,1]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if } \ldots \ | \ (\mu X)\, r\,[4,1]?_w \mathit{Accept} \cdots \oplus \mathit{Restart}.X \oplus \mathit{Abort}.\mathit{end})$
$| \ ((\mu X)\, s\,[4,2]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if } \ldots \ | \ (\mu X)\, r\,[4,2]?_w \mathit{Accept} \cdots \oplus \mathit{Restart}.X \oplus \mathit{Abort}.\mathit{end})$
$| \ ((\mu X)\, s\,[4,3]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if } \ldots \ | \ (\mu X)\, r\,[3,4]!_u l1b\,\langle\text{Promise Nothing}\rangle\,.\,\mathrm{P}^{\mathrm{a}}_{\mathrm{cont}})$
$| \ \Pi_{1\le k,l\le 4, k\ne l}\, s_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 4, k\ne l}\, r_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 2, k\ne l}\, t_{k\to l} : [])$

$\longmapsto (\mathit{USend} + \mathit{UGet})$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[4,1]!_u l1a\,\langle\text{proposalNumber}\,(5,4)\rangle\,.s\,[4,2]!_u l1a\,\langle\text{proposalNumber}\,(5,4)\rangle \ldots$
$| \ (\mu X)\,\text{if false or } 2 < 1\,\text{then} \ \ldots\text{else} \ r\,[4,\{2,3\}]!_w \mathit{Accept}.\ldots$
$| \ ((\mu X)\, s\,[4,1]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if } \ldots \ | \ (\mu X)\, r\,[4,1]?_w \mathit{Accept} \cdots \oplus \mathit{Restart}.X \oplus \mathit{Abort}.\mathit{end})$
$| \ ((\mu X)\, s\,[4,2]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if } \ldots \ | \ (\mu X)\, r\,[4,2]?_w \mathit{Accept} \cdots \oplus \mathit{Restart}.X \oplus \mathit{Abort}.\mathit{end})$
$| \ ((\mu X)\, s\,[4,3]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if } \ldots \ | \ (\mu X)\, r\,[4,3]?_w \mathit{Accept} \cdots \oplus \mathit{Restart}.X \oplus \mathit{Abort}.\mathit{end})$
$| \ \Pi_{1\le k,l\le 4, k\ne l}\, s_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 4, k\ne l}\, r_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 2, k\ne l}\, t_{k\to l} : [])$

$=$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[4,1]!_u l1a\,\langle\text{proposalNumber}\,(5,4)\rangle\,.s\,[4,2]!_u l1a\,\langle\text{proposalNumber}\,(5,4)\rangle \ldots$
$| \ (\mu X)\, r\,[4,\{2,3\}]!_w \mathit{Accept}.r\,[4,2]!_u l2a\left\langle\text{Proposal proposalNumber}\,(5,6)\ \text{promiseValue}\left(\overrightarrow{V}\right)\right\rangle \ldots$
$| \ ((\mu X)\, s\,[4,1]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if } \ldots \ | \ (\mu X)\, r\,[4,1]?_w \mathit{Accept} \cdots \oplus \mathit{Restart}.X \oplus \mathit{Abort}.\mathit{end})$
$| \ ((\mu X)\, s\,[4,2]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if } \ldots \ | \ (\mu X)\, r\,[4,2]?_w \mathit{Accept} \cdots \oplus \mathit{Restart}.X \oplus \mathit{Abort}.\mathit{end})$
$| \ ((\mu X)\, s\,[4,3]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if } \ldots \ | \ (\mu X)\, r\,[4,3]?_w \mathit{Accept} \cdots \oplus \mathit{Restart}.X \oplus \mathit{Abort}.\mathit{end})$
$| \ \Pi_{1\le k,l\le 4, k\ne l}\, s_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 4, k\ne l}\, r_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 2, k\ne l}\, t_{k\to l} : [])$

$\longmapsto (\mathit{WSel} + \mathit{WBran})$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[4,1]!_u l1a\,\langle\text{proposalNumber}\,(5,4)\rangle\,.s\,[4,2]!_u l1a\,\langle\text{proposalNumber}\,(5,4)\rangle \ldots$
$| \ (\mu X)\, r\,[4,2]!_u l2a\left\langle\text{Proposal proposalNumber}\,(5,6)\ \text{promiseValue}\left(\overrightarrow{V}\right)\right\rangle.$
$\quad r\,[4,3]!_u l2a\left\langle\text{Proposal proposalNumber}\,(5,6)\ \text{promiseValue}\left(\overrightarrow{V}\right)\right\rangle \ldots$
$| \ ((\mu X)\, s\,[4,1]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if } \ldots \ | \ (\mu X)\, r\,[4,1]?_w \mathit{Accept} \cdots \oplus \mathit{Restart}.X \oplus \mathit{Abort}.\mathit{end})$
$| \ ((\mu X)\, s\,[4,2]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if } \ldots \ | \ (\mu X)\, r\,[4,2]?_u l2a\,\langle\bot\rangle\,(pr')\,.\,\text{if } \ldots)$
$| \ ((\mu X)\, s\,[4,3]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if } \ldots \ | \ (\mu X)\, r\,[4,3]?_u l2a\,\langle\bot\rangle\,(pr')\,.\,\text{if } \ldots)$
$| \ \Pi_{1\le k,l\le 4, k\ne l}\, s_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 4, k\ne l}\, r_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 2, k\ne l}\, t_{k\to l} : [])$

$\longmapsto (\mathit{WSkip})$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[4,1]!_u l1a\,\langle\text{proposalNumber}\,(5,4)\rangle\,.s\,[4,2]!_u l1a\,\langle\text{proposalNumber}\,(5,4)\rangle \ldots$

$$| \ (\mu X)\, r\,[4,2]!_u l2a \left\langle \text{Proposal proposalNumber}\,(5,6)\ \text{promiseValue}\left(\overrightarrow{V}\right)\right\rangle .$$
$$r\,[4,3]!_u l2a \left\langle \text{Proposal proposalNumber}\,(5,6)\ \text{promiseValue}\left(\overrightarrow{V}\right)\right\rangle \ldots$$
$$| \ ((\mu X)\, s\,[4,1]?_u l1a\, \langle\perp\rangle\, (n')\,.\,\text{if } \ldots \ | \ (\mu X)\, \mathit{end})$$
$$| \ ((\mu X)\, s\,[4,2]?_u l1a\, \langle\perp\rangle\, (n')\,.\,\text{if } \ldots \ | \ (\mu X)\, r\,[4,2]?_u l2a\, \langle\perp\rangle\, (pr')\,.\,\text{if } \ldots)$$
$$| \ ((\mu X)\, s\,[4,3]?_u l1a\, \langle\perp\rangle\, (n')\,.\,\text{if } \ldots \ | \ (\mu X)\, r\,[4,3]?_u l2a\, \langle\perp\rangle\, (pr')\,.\,\text{if } \ldots)$$
$$| \ \Pi_{1\le k,l\le 4,k\ne l}\ s_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 4,k\ne l}\ r_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 2,k\ne l}\ t_{k\to l} : []\big)$$

$$\longmapsto (USend + UGet)$$
$$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[4,1]!_u l1a\, \langle \text{proposalNumber}\,(5,4)\rangle\, .s\,[4,2]!_u l1a\, \langle \text{proposalNumber}\,(5,4)\rangle \ldots$$
$$| \ (\mu X)\, r\,[4,3]!_u l2a \left\langle \text{Proposal proposalNumber}\,(5,6)\ \text{promiseValue}\left(\overrightarrow{V}\right)\right\rangle .\mathit{end}$$
$$| \ (\mu X)\, s\,[4,1]?_u l1a\, \langle\perp\rangle\, (n')\,.\,\text{if } \ldots$$
$$| \ ((\mu X)\, s\,[4,2]?_u l1a\, \langle\perp\rangle\, (n')\,.\,\text{if } \ldots \ | \ (\mu X)\, \text{if Proposal 10 abc} =\perp \text{ then } X \text{ else } \ldots)$$
$$| \ ((\mu X)\, s\,[4,3]?_u l1a\, \langle\perp\rangle\, (n')\,.\,\text{if } \ldots \ | \ (\mu X)\, r\,[4,3]?_u l2a\, \langle\perp\rangle\, (pr')\,.\,\text{if } \ldots)$$
$$| \ \Pi_{1\le k,l\le 4,k\ne l}\ s_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 4,k\ne l}\ r_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 2,k\ne l}\ t_{k\to l} : []\big)$$

$$\longmapsto (USend + UGet)$$
$$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[4,1]!_u l1a\, \langle \text{proposalNumber}\,(5,4)\rangle\, .s\,[4,2]!_u l1a\, \langle \text{proposalNumber}\,(5,4)\rangle \ldots$$
$$| \ (\mu X)\, \mathit{end}$$
$$| \ (\mu X)\, s\,[4,1]?_u l1a\, \langle\perp\rangle\, (n')\,.\,\text{if } \ldots$$
$$| \ ((\mu X)\, s\,[4,2]?_u l1a\, \langle\perp\rangle\, (n')\,.\,\text{if } \ldots \ | \ (\mu X)\, \text{if Proposal 10 abc} =\perp \text{ then } X \text{ else } \ldots)$$
$$| \ ((\mu X)\, s\,[4,3]?_u l1a\, \langle\perp\rangle\, (n')\,.\,\text{if } \ldots \ | \ (\mu X)\, \text{if Proposal 10 abc} =\perp \text{ then } X \text{ else } \ldots)$$
$$| \ \Pi_{1\le k,l\le 4,k\ne l}\ s_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 4,k\ne l}\ r_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 2,k\ne l}\ t_{k\to l} : []\big)$$

$$=$$
$$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[4,1]!_u l1a\, \langle \text{proposalNumber}\,(5,4)\rangle\, .s\,[4,2]!_u l1a\, \langle \text{proposalNumber}\,(5,4)\rangle \ldots$$
$$| \ (\mu X)\, s\,[4,1]?_u l1a\, \langle\perp\rangle\, (n')\,.\,\text{if } \ldots$$
$$| \ ((\mu X)\, s\,[4,2]?_u l1a\, \langle\perp\rangle\, (n')\,.\,\text{if } \ldots$$
$$| \ (\mu X)\, \text{if ge}\,(10,10)\,\text{then update}\,(pr,pr')\,.\,\text{update}\,(n,\text{Just } 10)\,.X \text{ else } \ldots)$$
$$| \ ((\mu X)\, s\,[4,3]?_u l1a\, \langle\perp\rangle\, (n')\,.\,\text{if } \ldots$$
$$| \ (\mu X)\, \text{if ge}\,(10,10)\,\text{then update}\,(pr,pr')\,.\,\text{update}\,(n,\text{Just } 10)\,.X \text{ else } \ldots)$$
$$| \ \Pi_{1\le k,l\le 4,k\ne l}\ s_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 4,k\ne l}\ r_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 2,k\ne l}\ t_{k\to l} : []\big)$$

$$=$$
$$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[4,1]!_u l1a\, \langle \text{proposalNumber}\,(5,4)\rangle\, .s\,[4,2]!_u l1a\, \langle \text{proposalNumber}\,(5,4)\rangle \ldots$$
$$| \ (\mu X)\, s\,[4,1]?_u l1a\, \langle\perp\rangle\, (n')\,.\,\text{if } \ldots$$
$$| \ ((\mu X)\, s\,[4,2]?_u l1a\, \langle\perp\rangle\, (n')\,.\,\text{if } \ldots \ | \ (\mu X)\, X)$$
$$| \ ((\mu X)\, s\,[4,3]?_u l1a\, \langle\perp\rangle\, (n')\,.\,\text{if } \ldots \ | \ (\mu X)\, X)$$
$$| \ \Pi_{1\le k,l\le 4,k\ne l}\ s_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 4,k\ne l}\ r_{k\to l} : [] \ | \ \Pi_{1\le k,l\le 2,k\ne l}\ t_{k\to l} : []\big)$$

$=$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\,s\,[4,1]!_u l1a\,\langle \text{proposalNumber}\,(5,4)\rangle\,.s\,[4,2]!_u l1a\,\langle \text{proposalNumber}\,(5,4)\rangle\ldots$
$\mid\ (\mu X)\,s\,[4,1]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots$
$\mid\ ((\mu X)\,s\,[4,2]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots\ \mid\ (\mu X)\,X)$
$\mid\ ((\mu X)\,s\,[4,3]?_u l1a\,\langle\perp\rangle\,(n')\,.\,\text{if}\ \ldots\ \mid\ (\mu X)\,X)$
$\mid \Pi_{1\le k,l\le 4,k\ne l}\ s_{k\to l} : [] \mid \Pi_{1\le k,l\le 4,k\ne l}\ r_{k\to l} : [] \mid \Pi_{1\le k,l\le 2,k\ne l}\ t_{k\to l} : []\big)$

$\longmapsto (\mathit{USend}+\mathit{UGet}, \mathit{USend}+\mathit{UGet}, \mathit{USkip})$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\,s\,[1,4]?_u l1b\,\langle\perp\rangle\,(v_1)\,.s\,[2,4]?_u l1b\,\langle\perp\rangle\,(v_2)\ldots$
$\mid\ (\mu X)\,\text{if}\ \ 5 =\perp\ \text{then}\ \ \text{P}^{\text{a}}_{\text{cont}}\ \text{else}\ \ldots$
$\mid\ ((\mu X)\,\text{if}\ \ 5 =\perp\ \text{then}\ \ \text{P}^{\text{a}}_{\text{cont}}\ \text{else}\ \ldots\ \mid\ (\mu X)\,X)$
$\mid\ ((\mu X)\,\text{if}\ \ \perp=\perp\ \text{then}\ \ \text{P}^{\text{a}}_{\text{cont}}\ \text{else}\ \ldots\ \mid\ (\mu X)\,X)$
$\mid \Pi_{1\le k,l\le 4,k\ne l}\ s_{k\to l} : [] \mid \Pi_{1\le k,l\le 4,k\ne l}\ r_{k\to l} : [] \mid \Pi_{1\le k,l\le 2,k\ne l}\ t_{k\to l} : []\big)$

$=$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\,s\,[1,4]?_u l1b\,\langle\perp\rangle\,(v_1)\,.s\,[2,4]?_u l1b\,\langle\perp\rangle\,(v_2)\ldots$
$\mid\ (\mu X)\,\text{if}\ \ \text{gt}\,(5,\text{Nothing})\,\text{then}\ \ \text{update}\,(n,5)\ldots\text{else}\ \ldots$
$\mid\ ((\mu X)\,\text{if}\ \ \text{gt}\,(5,\text{Just}\ \ 10)\,\text{then}\ \ \ldots\text{else}\ \ s\,[2,4]!_u l1b\,\langle\text{Nack}\ \ 10\rangle\,.\,\text{P}^{\text{a}}_{\text{cont}}\ \mid\ (\mu X)\,X)$
$\mid\ ((\mu X)\,r\,[4,3]?_w \mathit{Accept}\cdots\oplus \mathit{Restart}.X \oplus \mathit{Abort.end} \mid\ (\mu X)\,X)$
$\mid \Pi_{1\le k,l\le 4,k\ne l}\ s_{k\to l} : [] \mid \Pi_{1\le k,l\le 4,k\ne l}\ r_{k\to l} : [] \mid \Pi_{1\le k,l\le 2,k\ne l}\ t_{k\to l} : []\big)$

$=$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\,s\,[1,4]?_u l1b\,\langle\perp\rangle\,(v_1)\,.s\,[2,4]?_u l1b\,\langle\perp\rangle\,(v_2)\ldots$
$\mid\ (\mu X)\,s\,[1,4]!_u l1b\,\langle\text{Promise}\ \ \text{Nothing}\rangle\,.\,\text{P}^{\text{a}}_{\text{cont}}$
$\mid\ ((\mu X)\,s\,[2,4]!_u l1b\,\langle\text{Nack}\ \ 10\rangle\,.\,\text{P}^{\text{a}}_{\text{cont}}\ \mid\ (\mu X)\,X)$
$\mid\ ((\mu X)\,r\,[4,3]?_w \mathit{Accept}\cdots\oplus \mathit{Restart}.X \oplus \mathit{Abort.end} \mid\ (\mu X)\,X)$
$\mid \Pi_{1\le k,l\le 4,k\ne l}\ s_{k\to l} : [] \mid \Pi_{1\le k,l\le 4,k\ne l}\ r_{k\to l} : [] \mid \Pi_{1\le k,l\le 2,k\ne l}\ t_{k\to l} : []\big)$

$\longmapsto (\mathit{USend}+\mathit{UGet}, \mathit{USend}+\mathit{UGet})$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\,\text{if}\ \ \text{true}\ \ \textsf{or}\ 1<1\,\text{then}\ \ s\,[4,\{1,2\}]!_w \mathit{Restart}.X\ \text{else}\ \ldots$
$\mid\ (\mu X)\,s\,[4,1]?_w \mathit{Accept}\cdots\oplus \mathit{Restart}.X \oplus \mathit{Abort.end}$
$\mid\ ((\mu X)\,s\,[4,1]?_w \mathit{Accept}\cdots\oplus \mathit{Restart}.X \oplus \mathit{Abort.end} \mid\ (\mu X)\,X)$
$\mid\ ((\mu X)\,r\,[4,3]?_w \mathit{Accept}\cdots\oplus \mathit{Restart}.X \oplus \mathit{Abort.end} \mid\ (\mu X)\,X)$
$\mid \Pi_{1\le k,l\le 4,k\ne l}\ s_{k\to l} : [] \mid \Pi_{1\le k,l\le 4,k\ne l}\ r_{k\to l} : [] \mid \Pi_{1\le k,l\le 2,k\ne l}\ t_{k\to l} : []\big)$

$=$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\,s\,[4,\{1,2\}]!_w \mathit{Restart}.X$
$\mid\ (\mu X)\,s\,[4,1]?_w \mathit{Accept}\cdots\oplus \mathit{Restart}.X \oplus \mathit{Abort.end}$
$\mid\ ((\mu X)\,s\,[4,1]?_w \mathit{Accept}\cdots\oplus \mathit{Restart}.X \oplus \mathit{Abort.end} \mid\ (\mu X)\,X)$

$\mid\ ((\mu X)\, r\,[4,3]?_w Accept \cdots \oplus Restart.X \oplus Abort.end \mid\ (\mu X)\, X)$
$\mid \Pi_{1\leq k,l\leq 4, k\neq l}\ s_{k\to l} : [] \mid \Pi_{1\leq k,l\leq 4, k\neq l}\ r_{k\to l} : [] \mid \Pi_{1\leq k,l\leq 2, k\neq l}\ t_{k\to l} : []\big)$

$\longmapsto (WSel + WBran, WSkip)$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, X$
$\mid\ (\mu X)\, X$
$\mid\ ((\mu X)\, X \mid\ (\mu X)\, X)$
$\mid\ ((\mu X)\, end \mid\ (\mu X)\, X)$
$\mid \Pi_{1\leq k,l\leq 4, k\neq l}\ s_{k\to l} : [] \mid \Pi_{1\leq k,l\leq 4, k\neq l}\ r_{k\to l} : [] \mid \Pi_{1\leq k,l\leq 2, k\neq l}\ t_{k\to l} : []\big)$

$=$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[4,1]!_u l1a\,\langle \text{proposalNumber}\,(6,4)\rangle\,.s\,[4,2]!_u l1a\,\langle \text{proposalNumber}\,(6,4)\rangle \ldots$
$\mid\ (\mu X)\, s\,[4,1]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if } \ldots$
$\mid\ ((\mu X)\, s\,[4,2]?_u l1a\,\langle\bot\rangle\,(n')\,.\,\text{if } \ldots \mid\ (\mu X)\, X)$
$\mid\ (\mu X)\, X$
$\mid \Pi_{1\leq k,l\leq 4, k\neq l}\ s_{k\to l} : [] \mid \Pi_{1\leq k,l\leq 4, k\neq l}\ r_{k\to l} : [] \mid \Pi_{1\leq k,l\leq 2, k\neq l}\ t_{k\to l} : []\big)$

$\longmapsto (USend + UGet, USend + UGet)$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[1,4]?_u l1b\,\langle\bot\rangle\,(v_1)\,.s\,[2,4]?_u l1b\,\langle\bot\rangle\,(v_2)\,.\,\text{if } \ldots$
$\mid\ (\mu X)\,\text{if } 15 =\bot \text{ then } \mathrm{P}^{\mathrm{a}}_{\mathrm{cont}}\text{ else } \ldots$
$\mid\ ((\mu X)\,\text{if } 15 =\bot \text{ then } \mathrm{P}^{\mathrm{a}}_{\mathrm{cont}}\text{ else } \ldots \mid\ (\mu X)\, X)$
$\mid\ (\mu X)\, X$
$\mid \Pi_{1\leq k,l\leq 4, k\neq l}\ s_{k\to l} : [] \mid \Pi_{1\leq k,l\leq 4, k\neq l}\ r_{k\to l} : [] \mid \Pi_{1\leq k,l\leq 2, k\neq l}\ t_{k\to l} : []\big)$

$=$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[1,4]?_u l1b\,\langle\bot\rangle\,(v_1)\,.s\,[2,4]?_u l1b\,\langle\bot\rangle\,(v_2)\,.\,\text{if } \ldots$
$\mid\ (\mu X)\,\text{if gt}\,(15, \text{Just } 5)\,\text{then update}\,(n, 15)\ldots\text{else } \ldots$
$\mid\ ((\mu X)\,\text{if gt}\,(15, \text{Just } 10)\,\text{then update}\,(n, 15)\ldots\text{else } \ldots \mid\ (\mu X)\, X)$
$\mid\ (\mu X)\, X$
$\mid \Pi_{1\leq k,l\leq 4, k\neq l}\ s_{k\to l} : [] \mid \Pi_{1\leq k,l\leq 4, k\neq l}\ r_{k\to l} : [] \mid \Pi_{1\leq k,l\leq 2, k\neq l}\ t_{k\to l} : []\big)$

$=$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[1,4]?_u l1b\,\langle\bot\rangle\,(v_1)\,.s\,[2,4]?_u l1b\,\langle\bot\rangle\,(v_2)\,.\,\text{if } \ldots$
$\mid\ (\mu X)\, s\,[1,4]!_u l1b\,\langle\text{Promise Nothing}\rangle\,.\,\mathrm{P}^{\mathrm{a}}_{\mathrm{cont}}$
$\mid\ ((\mu X)\, s\,[2,4]!_u l1b\,\langle\text{Promise Proposal } 10\ abc\rangle\,.\,\mathrm{P}^{\mathrm{a}}_{\mathrm{cont}} \mid\ (\mu X)\, X)$
$\mid\ (\mu X)\, X$
$\mid \Pi_{1\leq k,l\leq 4, k\neq l}\ s_{k\to l} : [] \mid \Pi_{1\leq k,l\leq 4, k\neq l}\ r_{k\to l} : [] \mid \Pi_{1\leq k,l\leq 2, k\neq l}\ t_{k\to l} : []\big)$

$\longmapsto (USend + UGet, USend + UGet)$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\,\text{if false or } 2 < 1\,\text{then } \ldots\text{else } s\,[4,\{1,2\}]!_w Accept.\ldots$
$\mid\ (\mu X)\, s\,[4,1]?_w Accept \cdots \oplus Restart.X \oplus Abort.end$

$| \ ((\mu X)\, s\,[4,1]?_w Accept \cdots \oplus Restart.X \oplus Abort.end \mid \ (\mu X)\, X)$
$| \ (\mu X)\, X$
$| \ \Pi_{1\le k,l\le 4,k\neq l}\ s_{k\to l} : [] \mid \Pi_{1\le k,l\le 4,k\neq l}\ r_{k\to l} : [] \mid \Pi_{1\le k,l\le 2,k\neq l}\ t_{k\to l} : [])$

$=$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[4,\{1,2\}]!_w Accept. \ldots$
$| \ (\mu X)\, s\,[4,1]?_w Accept \cdots \oplus Restart.X \oplus Abort.end$
$| \ ((\mu X)\, s\,[4,1]?_w Accept \cdots \oplus Restart.X \oplus Abort.end \mid \ (\mu X)\, X)$
$| \ (\mu X)\, X$
$| \ \Pi_{1\le k,l\le 4,k\neq l}\ s_{k\to l} : [] \mid \Pi_{1\le k,l\le 4,k\neq l}\ r_{k\to l} : [] \mid \Pi_{1\le k,l\le 2,k\neq l}\ t_{k\to l} : [])$

$\longmapsto (WSel + WBran)$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, s\,[4,1]!_u l2a \left\langle \text{Proposal proposalNumber}\,(6,4)\ \text{promiseValue}\left(\overrightarrow{V}\right)\right\rangle .$
$s\,[4,2]!_u l2a \left\langle \text{Proposal proposalNumber}\,(6,4)\ \text{promiseValue}\left(\overrightarrow{V}\right)\right\rangle .end$
$| \ (\mu X)\, s\,[4,1]?_u l2a\,\langle\perp\rangle\,(pr')\,.\,\text{if} \ \ldots$
$| \ ((\mu X)\, s\,[4,2]?_u l2a\,\langle\perp\rangle\,(pr')\,.\,\text{if} \ \ldots \ \mid \ (\mu X)\, X)$
$| \ (\mu X)\, X$
$| \ \Pi_{1\le k,l\le 4,k\neq l}\ s_{k\to l} : [] \mid \Pi_{1\le k,l\le 4,k\neq l}\ r_{k\to l} : [] \mid \Pi_{1\le k,l\le 2,k\neq l}\ t_{k\to l} : [])$

$\longmapsto (USend + UGet, USend + UGet)$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\,(\mu X)\, end$
$| \ (\mu X)\,\text{if Proposal 15 Proposal 15}\ abc = \perp \text{ then } \ldots \text{else } \ldots$
$| \ ((\mu X)\,\text{if Proposal 15 Proposal 15}\ abc = \perp \text{ then } \ldots \text{else } \ldots \ \mid \ (\mu X)\, X)$
$| \ (\mu X)\, X$
$| \ \Pi_{1\le k,l\le 4,k\neq l}\ s_{k\to l} : [] \mid \Pi_{1\le k,l\le 4,k\neq l}\ r_{k\to l} : [] \mid \Pi_{1\le k,l\le 2,k\neq l}\ t_{k\to l} : [])$

$=$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\mid \ (\mu X)\,\text{if ge}\,(15, \text{Just } 15)$
$\quad \text{then update}\,(pr, \text{Proposal } 15\ abc)\,.\,\text{update}\,(n,15)\,.X \text{ else } \ldots$
$| \ ((\mu X)\,\text{if ge}\,(15, \text{Just } 15)$
$\quad \text{then update}\,(pr, \text{Proposal } 15\ abc)\,.\,\text{update}\,(n,15)\,.X \text{ else } \ldots \ \mid \ (\mu X)\, X)$
$| \ (\mu X)\, X$
$| \ \Pi_{1\le k,l\le 4,k\neq l}\ s_{k\to l} : [] \mid \Pi_{1\le k,l\le 4,k\neq l}\ r_{k\to l} : [] \mid \Pi_{1\le k,l\le 2,k\neq l}\ t_{k\to l} : [])$

$=$
$(\nu t)\,(\nu s)\,(\nu r)\,\big(\mid \ (\mu X)\, X$
$| \ ((\mu X)\, X \mid \ (\mu X)\, X)$
$| \ (\mu X)\, X$
$| \ \Pi_{1\le k,l\le 4,k\neq l}\ s_{k\to l} : [] \mid \Pi_{1\le k,l\le 4,k\neq l}\ r_{k\to l} : [] \mid \Pi_{1\le k,l\le 2,k\neq l}\ t_{k\to l} : [])$

$$
\begin{aligned}
&\longmapsto (USkip + WSkip, USkip + WSkip, USkip + WSkip, USkip + WSkip)\\
&(\nu t)\,(\nu s)\,(\nu r)\,\big(\mid\ (\mu X)\, end\\
&\mid\ ((\mu X)\, end \mid\ (\mu X)\, end)\\
&\mid\ (\mu X)\, end\\
&\mid \Pi_{1\le k,l\le 4,k\ne l}\ s_{k\to l} : [] \mid \Pi_{1\le k,l\le 4,k\ne l}\ r_{k\to l} : [] \mid \Pi_{1\le k,l\le 2,k\ne l}\ t_{k\to l} : []\big)\\
&=\\
&(\nu t)\,(\nu s)\,(\nu r)\,(\ \mid \Pi_{1\le k,l\le 4,k\ne l}\ s_{k\to l} : [] \mid \Pi_{1\le k,l\le 4,k\ne l}\ r_{k\to l} : [] \mid \Pi_{1\le k,l\le 2,k\ne l}\ t_{k\to l} : [])
\end{aligned}
$$

4 Evaluation

RESULTS

5 Discussion

DISCUSSION