Nominated for 1998 IEEE/ACM/SCS PADS Best Paper Award

# Performance Analysis of Time Warp Simulation with Cascading Rollbacks

*Seng Chuan TAY, Yong Meng TEO* and *Rassul Ayani* *
Department of Information Systems & Computer Science
National University of Singapore
Lower Kent Ridge Road
Singapore 119260
email: taysengc@iscs.nus.edu.sg

## Abstract

*This paper presents an analytical model for evaluating the performance of Time Warp simulators. The proposed model is formalized based on two important time components in parallel and distributed processing: computation time and communication time. The communication time is modeled by buffer access time and message transmission time. Logical processes of the Time Warp simulation, and the processors executing them are assumed to be homogeneous. Performance metrics such as rollback probability, rollback distance, elapsed time and Time Warp efficiency are derived. More importantly, we also analyze the impact of cascading rollback waves on the overall Time Warp performance. By rendering the deviation in state numbers of sender-receiver pairs, we investigate the performance of throttled Time Warp scheme. Our analytical model shows that the deviation in state numbers and the communication delay have a profound impact on Time Warp efficiency. The performance model has been validated against implementation results obtained on a Fujitsu AP3000 parallel computer. The analytical framework can be readily used to estimate performance before the Time Warp simulator is implemented.*

## 1 Introduction

Although Time Warp (TW) mechanism [5] can potentially exploit a higher degree of parallelism [13] in the simulated system, runaway parallelism can result in unnecessary rollback thrashing. Such an ill effect is also aggravated on parallel or distributed computing platforms where long and wide variation in communication delays cause a broad spread of individual simulation progress. The event messages may arrive untimely. In the worst case, most of the time is spent on performing rollback so called rollback thrashing. Due to this unstable phenomenon, the performance of TW simulator is often difficult to predict whenever the modeling parameters such as arrival rate and duration of simulation are changed. Most of the existing performance studies are either confined in a two-processor configuration, or ignore communication overhead. Some models also assume negligible state-saving and rollback costs [8]. Such assumptions, however, are invalid in most practical cases.

The proposed model is based on more realistic approximations, e.g., it includes communication overhead, state-saving cost and cascading rollback. Hence, the analytical model provides a practical framework to characterize the performance of TW. Input parameters of the model, such as event and state saving time, buffer access time and message transmission time, arrival and service rates, can be estimated using a sequential simulator (or a parallel simulator similar to the one that should be designed) and specifications of the parallel computer that will be used. Thus, our analytical model can be used to predict TW simulation performance before it is implemented. We model the deviation in LP state numbers of each sender-receiver pair using a normalized statistical distribution[1], and develop a method to analyze the cost of cascading rollback. By rendering the statistical distribution, the proposed scheme can be extended to model and optimize the performance of throttled TW.

The rest of this paper is organized as follows. Section 2 gives an overview of related work on performance modeling of TW. We highlight the coverage and limitation of the existing models. Section 3 describes a parallelism throttle used to control rollback thrashing. Section 4 adopts a probabilistic approach to model the LVT advancement in LPs. We derive the *rollback probability* and *rollback distance* caused by untimely event arrivals, the simulator *elapsed time*, and analyze the TW *efficiency*. In addition, we also propose a method used to analyze the overhead incurred by cascading rollbacks. Section 5 analyzes the stability of the perfor-

[*] Rassul Ayani is a professor of computer science at the Royal Institute of Technology in Stockholm, Sweden. He is currently a visiting professor in the Department of Information Systems and Computer Science at the National University of Singapore.

[1] The normalized statistical distribution ($f_{N_w}$), is the discrete version of the continuous normal distribution, where $w$ is the mode of the state number. The discrete distribution has a parameter called spread ($\chi$), that is similar to the standard deviation in the continuous normal distribution.

1087-4097/98 $10.00 © 1998 IEEE

mance model, and compares the derived metrics from the proposed analytical performance model with results from the parallel simulation of a multi-stage interconnection network (MIN). Using the performance model, we also analyze the sensitivity of the state proximity on TW efficiency, and illustrate the use of parallelism throttle to improve the performance. Lastly, section 6 contains our concluding remarks.

## 2 Overview of Related Work

The earliest work on TW performance modeling has been devoted to two-processor systems [6, 7, 10, 12]. Some performance models do not consider communication and synchronization overheads. Lin and Lazowska [8] use the critical-path analysis to show that TW always performs at least as well as conservative methods if the state-saving and rollback costs are negligible. They show that if the simulator rolls back only false computations, TW performance will be optimal. Nicol derives the upper bound of TW performance using self-initiating model [11]. His analysis includes the state saving and rollback costs, however, the effect of cascading rollback is ignored. Felderman et. al. also derive performance bounds for self-initiating model but do not include the state saving and rollback cost [3]. They assume an arbitrary number of processors and a uniform connection topology. By tracking the progression of global virtual time, they provide the upper and lower bounds on the performance speedup for optimistic simulation.

Another category of TW performance models is based on Markov-chains analysis. Gupta et. al. use Markov chains to model the performance of TW for multiple homogeneous processors [4]. The model assumes that the number of unprocessed event messages is a constant throughout the simulation. Performance results are then approximated using numerical methods. Communication delay, state saving and rollback costs are also assumed to be negligible in their model. Using the Markov-chains approach to analyze TW augmented with a cancelback protocol, the effect of memory capacity on TW performance is studied by Akyildiz et. al. [1]. The TW simulator is assumed to run on a shared-memory multiprocessor, and the message population is fixed throughout the simulation. Their model is able to predict the speedup if the amount of memory is varied. They also show that if the sequential simulation require $m$ messages buffers, TW with a small fraction of message buffers beyond $m$ performs almost as well as TW under unlimited memory. Due to the increased computational complexity for large-scale model, performance models based on Markov chains are always simplified and the results are usually in terms of approximation.

Branch and bound method is used by Lubachevsky et. al. to analyze the stability of TW [9]. They use two parameters: branch factor ($b$) which is the average number of nodes that receive anti-messages, and bound factor ($h$) which is related to the rate at which information about incorrect events propagates through the system. The model derives a relationship that if $b < e^h$ the rollback-based simulation is efficient. Otherwise, the simulator is trapped in a cascading rollback.

Based on probabilistic approach and numerical techniques, Dickens et. al. analyze the performance of bounded TW [2]. Although their model includes the state-saving cost, the rollback cost is computed for stragglers only. No analysis is done on cascading rollback and communication delay.

The analytic approach used in this paper has some resemblance to the existing work. We adopt a probabilistic approach and focus on the performance modeling of the throttled TW scheme that constrains the degree of speculative parallelism allowed at runtime. Our model is based on the adherence to realistic assumptions and characterization of various overhead costs, namely non-negligible communication delay, state-saving cost, and cascading rollback, and is scalable to $n$ processors. In addition, we model the degree of speculative parallelism using the deviation of state numbers.

## 3 Throttled Time Warp

Briefly, the control scheme adopts a *two-sided* approach where slow LPs are accelerated, and fast LPs suspended [14]. Let $r$ be a *spread ratio* such that $0 \leq r \leq 1$[2], and GVT be the global virtual (slowest) time, and GFT the global furthest (fastest) time among all LPs.

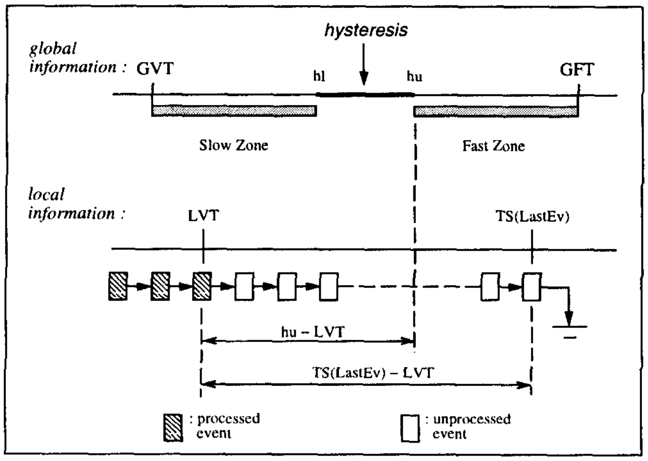

Figure 1: Using Local and Global Information to parameterize Event Regulator

We first divide the global progress window (GPW), denoted as [GVT, GFT], into three zones (see figure 1): *slow* [GVT, hl), *hysteresis* [hl, hu], *fast* (hu, GFT]. The hysteresis zone is bounded by $hl$ (lower bound) and $hu$ (upper bound), and centered in the work window. Accordingly, we have $\frac{hu+hl}{2} = \frac{GFT+GVT}{2}$. As $r$ determines the ratio of the length of the hysteresis zone to the work window, we have $hu - hl =$

[2] $r = 0$ indicates in-pace parallel simulation, and $r = 1$ indicates the worst (widest) disparity in LP advancement.

$r \times (GFT - GVT)$. Solving these two equations, we have $hl = (0.5 + r/2) \times GVT + (0.5 - r/2) \times GFT$, and $hu = (0.5 - r/2) \times GVT + (0.5 + r/2) \times GFT$. It is noteworthy that the computation of GFT (taking maximum, instead of the minimum used in GVT) can be embedded in the existing GVT acquisition protocol.

To prevent any racing phenomenon caused by uncontrolled LVT acceleration, an event execution regulator, denoted by $k$, is used to constrain the event execution speed. In our throttling scheme the acceleration is terminated when the LVT of slow LPs sweeps pass the upper bound of hysteresis zone. We also bound the regulator by a value $k_{max}$ so that the LVT acceleration does not exceed the capacity of communication channels. For each slow LP the regulation of event execution speed is based on its LVT position on GPW and the status of its input buffer. Let $TS(LastEv)$ be the timestamp of the last event message in the input queue. The parameterization used to control the acceleration is $k = k_{max} \times \min(\frac{TS(LastEv) - LVT}{hu - LVT}, 1)$, which allows more events to be executed in the LP cycle if the timestamp of the last pending event is further away from the LVT (figure 1). A new GPW is computed when the LVT in the first two types of LP passes the hysteresis zone so that the fast LPs are able to resume their event execution.

## 4 Analytic Performance Model

Table 1 contains a list of performance parameters used in the proposed probabilistic model. We assume that the TW simulator contains $p$ homogeneous LPs where each of them is executed by an exclusive homogeneous processor (PP). The distribution of state numbers is modeled by a normalized discrete probability density function with a parameter $\chi$ representing the deviation of state numbers [15]. We assume that the system state vector is saved after each event is executed. Each state is dynamically assigned an index based on its timestamp order. State indices are re-used after a rollback is activated but not after a GVT advancement.

Communication delay is modeled by two time components: *buffer access time* and *transmission time*. The buffer access time is accounted to both sending and receiving LPs. In order to prevent double accounting, the transmission time is accounted by the sender only. The wall clock duration for each transmission is $T_{buffer} + T_{transit}$, and for each reception is $T_{buffer}$ (figure 2). Therefore, the duration for a message to travel from its sender to the receiver is $2 \times T_{buffer} + T_{transit}$, and the number of events processed (and the number of states saved) during this communication delay is $c = \lceil \frac{2 \times T_{buffer} + T_{transit}}{T_{event} + T_{state}} \rceil$. In the following analysis, exponential distribution is used to model the interarrival time and service time due to its memory-less property. GVT re-computation is performed whenever a predefined number of events are processed. The number of events executed in between two GVT computations is denoted by $\widehat{GVT}$.

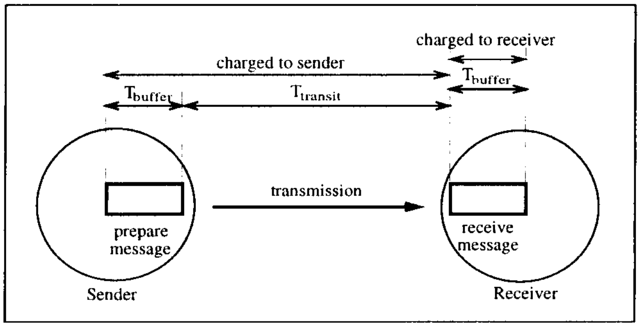

Figure 2: Communication Time Accounting

### 4.1 LVT Advancement Characterization

We assume that the inter-LVT advancement time has an exponential distribution of mean $\frac{1}{\beta}$, where $\beta$ is the LVT advancement rate defined as follows:

$$\beta = \begin{cases} 2\lambda & \text{if } \lambda < \mu \\ \lambda + \mu & \text{otherwise} \end{cases}$$

where $\lambda$ is the arrival rate, and $\mu$ the service rate. The LVT at the $n$-th state, denoted by $LVT_n$, $1 \leq n \leq N_p$, is modeled based on the following observations:

- The first event processed by an LP is an arrival event. Otherwise, the causality constraint is violated.
- An LP cannot advance its LVT until the first arrival event is processed.
- An LP at the $n$-th state has advanced its LVT $(n - 1)$ times.

Let $LVT_n$ denote the clock time in an LP when $n$ events are processed. Assume that the interarrival time and service time are identically and independently distributed (IID). We can parameterize $LVT_n$ as sum of two random variables $R_1$ and $R_2$, where $R_1 \sim \exp(\lambda)$, and $R_2 \sim$ gamma $(\beta, n - 1)$. Let random variable $Z$ represent the LVT of an LP at the $n$-th state. The probability density function of $Z$ is given as follows:

$$g(z) = \begin{cases} \lambda T^{n-1} \times \left(e^{-\lambda z} - e^{-\beta z} \sum_{k=0}^{n-2} \frac{(\theta z)^k}{k!}\right) & \text{if } z \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

where $\theta = \beta - \lambda$, and $T = \frac{\beta}{\theta}$ (refer to [15]).

### 4.2 Causality Error Characterization

Suppose an event message $\mathcal{M}$ is generated at the $p$-th state of the sending LP, and processed at the $q$-th state of the receiving LP. Let the timestamp of $\mathcal{M}$ be $LVT_{p,send}$, and the LVT of the target LP be $LVT_{q,recv}$. We want to compute $\Pr(LVT_{p,send} < LVT_{q,recv})$, which is the probability that $\mathcal{M}$ is out of sequence when it is processed by the receiving LP.

Let $X$ and $Y$ be random variables for $LVT_{p,send}$ and $LVT_{q,recv}$ respectively. Similarly, we can model

| parameter | | description |
|---|---|---|
| measured | $T_{event}$ | event (arrival or departure) execution time |
| | $T_{state}$ | state saving time |
| | $T_{buffer}$ | buffer (receive or transmit) access time |
| | $T_{transit}$ | message transmission time |
| system | $\lambda$ | arrival rate of each LP |
| | $\mu$ | service rate of each LP |
| | $\beta$ | LVT advancement rate |
| | $p$ | number of processors (PPs) |
| | $c$ | communication delay (in terms of number of events processed) |
| | $N_s$ | number of events processed in sequential simulation |
| | $N_p$ | number of true events processed by an LP ($N_p = \frac{N_s}{p}$) |
| | $\widehat{GVT}$ | number of events processed before a GVT computation is invoked |
| | $a$ | lower bound of GVT window (in terms of state index) |
| derived | $rb(I_0, J_0)$ | probability that an event message sent at state $I_0$ will cause a rollback when it is processed at state $J_0$ |
| | $RB(J_0)^+$ | probability that a straggler is processed at state $J_0$ |
| | $halt_{J_k}^{I_k}(d_k)$ | probability that a rollback caused by a straggler (or negative message) sent at state $I_k$ of the source LP and processed at state $J_k$ of target LP will stop after $d_k$ events are undone |
| | $D(J_0)^+$ | expected rollback distance (in terms of number of undone events) caused by the straggler processed at state $J_0$ |
| | $\overline{D^+}$ | expected rollback distance (due to straggler) within a GVT window |
| | $RB(J_n)_n^-$ | probability that an $n$-th wave of anti-messages is processed at state $J_n$ |
| | $D(J_n)_n^-$ | expected rollback distance (in terms of number of undone events) caused by the $n$-th wave of anti-messages processed at state $J_n$ |
| | $\overline{D_n^-}$ | expected rollback distance (due to the $n$-th wave of anti-message) within a GVT window |
| | $T_{Comp}$ | total computation time |
| | $T_{Comm}$ | total communication time |
| | $T_{TW}$ | elapsed time of TW simulator |
| | $\mathcal{E}$ | efficiency of TW simulator |

Table 1: Performance Model Parameters

$X$ by the statistical distributions exp($\lambda$) and gamma ($\beta$, $p-1$), and $Y$ by exp($\lambda$) and gamma ($\beta$, $q-1$). The probability density functions, denoted by $g_1(x)$ and $g_2(y)$ respectively are given as follows:

$$g_1(x) = \begin{cases} \lambda T^{p-1} \times \left(e^{-\lambda x} - e^{-\beta x} \sum_{k=0}^{p-2} \frac{(\theta x)^k}{k!}\right) & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

$$g_2(y) = \begin{cases} \lambda T^{q-1} \times \left(e^{-\lambda y} - e^{-\beta y} \sum_{l=0}^{q-2} \frac{(\theta y)^l}{l!}\right) & \text{if } y \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

Let $g_{X,Y}$ be the joint density function of $X$ and $Y$. We have

$$Pr(LVT_{p,send} < LVT_{q,recv}) = \int_0^\infty \int_0^y g_{X,Y}(x,y)\,dx\,dy.$$

Assume that $LVT_{p,send}$ and $LVT_{q,recv}$ are also IID. We can replace $g_{X,Y}(x,y)$ by $g_1(x) \times g_2(y)$. From [15],

$$Pr(LVT_{p,send} < LVT_{q,recv}) = \lambda^2 T^{p+q-2} \times$$
$$\left[\frac{1}{2\lambda^2} - \frac{G_p(\frac{\theta}{\lambda+\beta})}{\lambda(\lambda+\beta)} - \frac{G_q(\frac{\theta}{\beta})}{\lambda\beta} + \frac{G_q(\frac{\theta}{\lambda+\beta})}{\lambda(\lambda+\beta)} + \frac{G_p(\frac{\theta}{\beta}) \times G_q(\frac{\theta}{\beta})}{\beta^2}\right.$$
$$\left. - \frac{1}{2\beta^2} \sum_{l=0}^{q-2} \frac{(\frac{\theta}{2\beta})^l}{l!} \sum_{k=0}^{p-2} \left(\frac{\theta}{\beta}\right)^k \sum_{i=0}^{k} \left(\frac{(\frac{1}{2})^i \times (l+i)!}{i!}\right)\right]$$

where $G_n(x) = \sum_{i=0}^{n-2} x^i = \frac{1-x^{n-1}}{1-x}$.

### 4.2.1 Rollback Probability

For the ease of discussion we let $LP_0$ send an event message $\mathcal{M}$ to $LP_1$ (see figure 3). Let the index of $LP_0$ be $I_0$ when $\mathcal{M}$ is generated, and the index of $LP_1$ be $J_0$ when $\mathcal{M}$ is executed, where $0 \leq I_0, J_0 \leq N_p$. We observe that $\mathcal{M}$ will become a straggler in $LP_1$ if the timestamp of $\mathcal{M}$ (denoted by $LVT_{I_0,LP_0}$) is less than the LVT of the receiving LP at state $J_0$ (denoted by $LVT_{J_0,LP_1}$). As the progress of both LPs is bounded by a GVT window, we have $|I_0 - J_0| < \widehat{GVT}$. Due to the asynchronous event processing of each LP, $I_0$ can be of any value within the GVT window $[a, (a+\widehat{GVT}-1)]$, where $a$ is the lower bound (in terms of state index) of the window. Consider the homogeneity assumption imposed on all LPs and all PPs. If $LP_1$ processes $\mathcal{M}$ at state $J_0$, more likely (in terms of probability) the message is generated when $LP_0$ is at state $J_0 - c$, where $c$ is

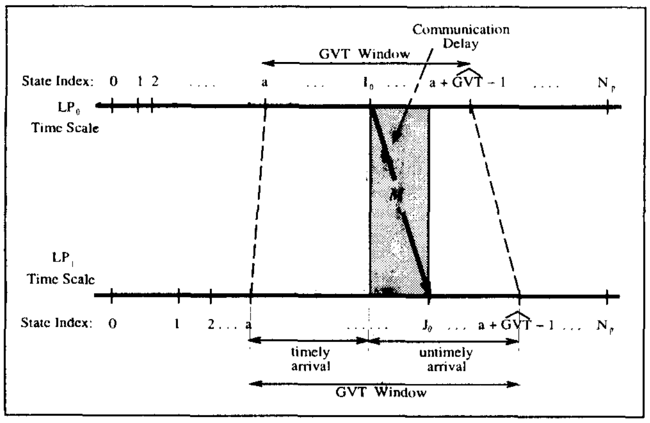
Communication Delay
GVT Window
State Index: 0 1 2 .... a ... I0 ... a + GVT̂ - 1 .... Np
LP0 Time Scale
M
LP1 Time Scale
State Index: 0 1 2 ... a ...... J0 ... a + GVT̂ - 1 ... Np
timely arrival
untimely arrival
GVT Window

Figure 3: Causality Error

the number of events processed by $LP_1$ during the communication delay. Within the GVT window the more likely state corresponds to the $(max(J_0 - c - a, 0))$-th event away from the lower bound. We therefore assign a normalized discrete distribution which is peaked at $max(J_0 - c - a, 0)$ to $I_0$ [15]. In the following derivation we let $I_0 = i_0 + a$, and $J_0 = j_0 + a$. It follows that $0 \leq i_0, j_0 \leq \widehat{GVT} - 1$.

Let $rb(I_0, J_0) = \Pr(LVT_{I_0,LP_0} < LVT_{J_0,LP_1})$, or $\Pr(LVT_{a+i_0,LP_0} < LVT_{J_0,LP_1})$ equivalently. The rollback probability (due to straggler) at state $J_0$ is

$$RB(J_0)^+ \quad = \quad \sum_{i_0=0}^{\widehat{GVT}-1} f_{N_{max(J_0-c-a,0)}}(i_0) \times rb(I_0, J_0).$$

### 4.2.2 Rollback Distance

Rollback distance is defined as the number of events to be undone. Suppose an LP at state $J_0$ receives a straggler sent by the preceding LP at state $I_0$ (see figure 4). We first compute $halt_{J_0}^{I_0}(d_0)$, which is the probability that the rollback will halt after $d_0$ events are undone, based on the following observations:

- $LVT_{I_0,LP_0} > LVT_{J_0-d_0,LP_1}$. Otherwise the rollback will not halt after $d_0$ events are undone.
- $LVT_{I_0,LP_0} < LVT_{J_0-d_0+1,LP_1}$. Otherwise the number of events undone is less than $d_0$.

As a rollback cannot coast below the lower bound of the GVT window, we impose the total probability constraint on $halt_{J_0}^{I_0}(d_0)$. In general, we ensure the condition $\sum_{d_k=1}^{J_k-a} halt_{J_k}^{I_k}(d_k) = 1$ for $k \geq 0$. This is done by normalizing the *halt* probability with respect to its sum as follows:

$$halt_{J_k}^{I_k}(d_k) \quad = \quad \frac{(1 - rb(I_k, J_k - d_k)) \times rb(I_k, J_k - d_k - 1)}{R\_SUM_{J_k}^{I_k}}$$

where $R\_SUM_{J_k}^{I_k} = \sum_{d_k=1}^{J_k-a}(1 - rb(I_k, J_k - d_k))\times$
$rb(I_k, J_k - d_k + 1)$.

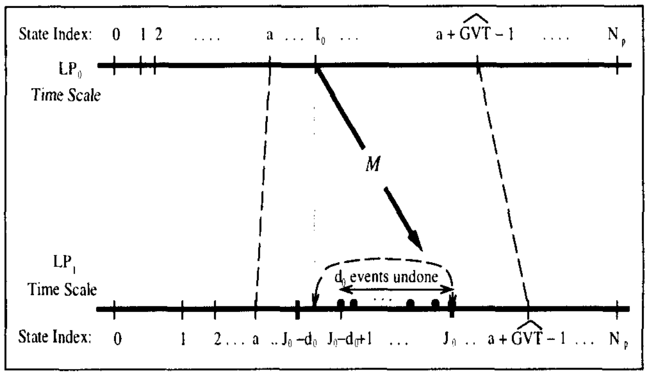
State Index: 0 1 2 .... a ... I0 ... a + GVT̂ - 1 .... Np
LP0 Time Scale
M
LP1 Time Scale
d0 events undone
State Index: 0 1 2 ... a .. J0 -d0 J0-d0+1 ... J0 .. a + GVT̂ - 1 ... Np

Figure 4: Rollback Distance due to Straggler

It follows that the expected rollback distance measured from state $J_0$ is $D(J_0)^+ = \sum_{i_0=0}^{\widehat{GVT}-1} \sum_{d_0=1}^{J_0-a} [d_0 \times f_{N_{max(J_0-c-a,0)}}(i_0) \times rb(I_0, J_0) \times halt_{J_0}^{I_0}(d_0)]$.

The expected rollback distance (due to straggler) within the GVT window is $D^+ = \sum_{J_0=0}^{\widehat{GVT}-1} \frac{D(J_0)^+}{\widehat{GVT}}$.

## 4.3 Cascading Rollback Characterization

Cascading rollback can be caused by several waves of negative messages. In the following analysis we derive the rollback probability and distance caused by the first wave of negative messages, and generalize the formulation for the subsequent waves.

### 4.3.1 First Wave

For the ease of description we let $LP_2$ be a successor of $LP_1$ (see figure 5). Assume that $LP_1$ rolls back $d_0$ events, and $w$ events in the rolled back interval, $w \leq d_0$, are sent to $LP_2$. During the rollback, negative messages will have to be sent to $LP_2$ to annihilate the side effects caused by such false events. We assume the worst case where the negative message, denoted by $\mathcal{M}^-$, corresponds to the positive copy sent at state $J_0 - d_0$. In the following analysis, $I_k$, $k \geq 1$, is used to index the $k$-th wave of negative message. We assume $I_k = J_{k-1} - d_{k-1}$ and, and let $I_k = i_k + a$, $0 \leq i_k \leq \widehat{GVT} - 1$.

#### Rollback Probability

If $LP_2$ receives $\mathcal{M}^-$ at state $J_1$, more likely the causality error occurs when $LP_1$ is at state $J_0 = J_1 - c$, and the straggler sent to $LP_1$ is generated when $LP_0$ is at state $I_0 = J_0 - c$. Again, we translate the more likely state indices within the GVT window, and assign the discrete distribution peaked at $max(J_0 - c - a, 0) = max(j_0 - c, 0)$ and $max(J_1 - c - a, 0) = max(j_1 - c, 0)$ to $I_0$ and $J_0$ respectively.

We observe the first wave of cascading rollback in $LP_2$ if the following conditions are fulfilled:

- The event message sent to $LP_1$ becomes a straggler in the target LP, i.e., $LVT_{I_0,LP_0} < LVT_{J_0,LP_1}$.
- When $\mathcal{M}^-$ arrives in $LP_2$, its positive copy of time stamp $LVT_{J_0-d_0,LP_1}$ (or $LVT_{I_1,LP_1}$ equivalently)

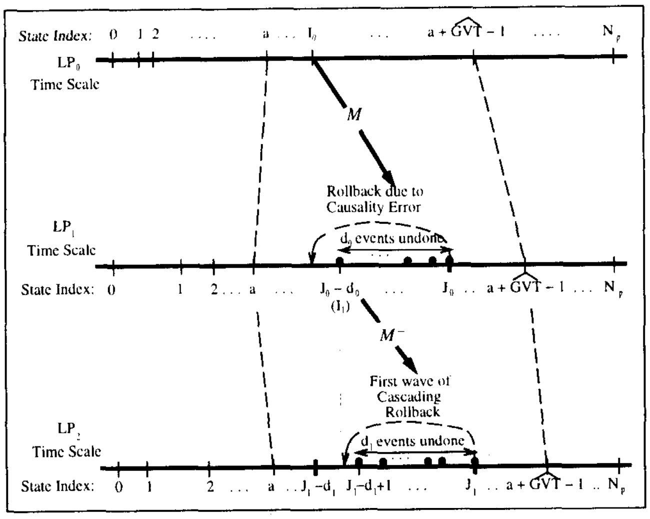
State Index: 0 1 2 .... a ... $I_0$ ... $a+\widehat{GVT}-1$ .... $N_p$
$LP_0$ Time Scale
$M$
Rollback due to Causality Error
$LP_1$ Time Scale
$d_0$ events undone
State Index: 0 1 2 ... a ... $J_0-d_0$ ($I_1$) ... $J_0$ .. $a+\widehat{GVT}-1$ ... $N_p$
$M^-$
First wave of Cascading Rollback
$LP_2$ Time Scale
$d_1$ events undone
State Index: 0 1 2 ... a .. $J_1-d_1$ $J_1-d_1+1$ ... $J_1$ .. $a+\widehat{GVT}-1$ .. $N_p$

Figure 5: Cascading Rollback

has been processed by $LP_2$, i.e., $LVT_{I_1,LP_1} < LVT_{J_1,LP_2}$.

Therefore, the cascading rollback probability (due to the first-wave of negative message) at state $J_1$ can be expressed as $RB(J_1)_1^- = \sum_{i_0=0}^{\widehat{GVT}-1} \sum_{j_0=0}^{\widehat{GVT}-1} \sum_{d_0=1}^{j_0}$

$$[f_{N_{max(j_0-c,0)}}(i_0) \times rb(I_0,J_0) \times f_{N_{max(j_1-c,0)}}(j_0) \times halt_{J_0}^{I_0}(d_0) \times rb(I_1,J_1)].$$

### Rollback Distance

As the progress of $LP_2$ is also bounded by the same GVT window, its state index can also be of any value in $[a, (a+\widehat{GVT}-1)]$ when the negative message is received. As $a$ is the lower bound of the GVT window, the possible cascading rollback distance for the first wave, denoted by $d_1$, cannot be larger than $(J_1 - a)$, or $j_1$ equivalently, events. Therefore, the rollback distance caused by the first wave of negative message at state $J_1$ is $D(J_1)_1^- = \sum_{i_0=0}^{\widehat{GVT}-1} \sum_{j_0=0}^{\widehat{GVT}-1} \sum_{d_0=1}^{j_0} \sum_{d_1=1}^{j_1}$

$$[d_1 \times f_{N_{max(j_0-c,0)}}(i_0) \times rb(I_0,J_0) \times f_{N_{max(j_1-c,0)}}(j_0) \times halt_{J_0}^{I_0}(d_0) \times rb(I_1,J_1) \times halt_{J_1}^{I_1}(d_1)].$$

The expected cascading rollback distance caused by the first wave of negative message is

$$\overline{D_1^-} = \sum_{j_1=0}^{\widehat{GVT}-1} \frac{D(J_1)_1^-}{\widehat{GVT}}.$$

### 4.3.2 Formulation for $n$-th Wave

The subsequent waves of cascading rollback can be formulated recursively in the same manner. To simplify the mathematical expressions, we let $J_k = j_k + a$, and use $\prod_{k=0}^{n-1}\left(\sum_{j_k=0}^{\widehat{GVT}-1}\sum_{d_k=1}^{j_k}\right)$ to represent $\sum_{j_0=0}^{\widehat{GVT}-1}\sum_{d_0=1}^{j_0}\sum_{j_1=0}^{\widehat{GVT}-1}\sum_{d_1=1}^{j_1}\cdots\sum_{j_{n-1}=0}^{\widehat{GVT}-1}\sum_{d_{n-1}=1}^{j_{n-1}}$.

### Rollback Probability

The rollback probability (due to the $n$-th wave of negative messages) at state $J_n$ (or $j_n + a$ equivalently) can be formulated as

$$RB(J_n)_n^- = \sum_{i_0=0}^{\widehat{GVT}-1} \prod_{k=0}^{n-1} \left(\sum_{j_k=0}^{\widehat{GVT}-1} \sum_{d_k=1}^{j_k}\right) [f_{N_{max(j_0-c,0)}}(i_0) \times rb(I_0,J_0) \times \prod_{k=1}^{n} \left(f_{N_{max(j_k-c,0)}(j_{k-1})} \times halt_{J_{k-1}}^{I_{k-1}}(d_{k-1}) \times rb(I_k,J_k)\right)]$$

### Rollback Distance

The expected rollback distance (caused by the $n$-th wave of negative messages) when an LP is at state $J_n$ can be formulated as

$$D(J_n)_n^- = \sum_{i_0=0}^{\widehat{GVT}-1} \prod_{k=0}^{n-1} \left(\sum_{j_k=0}^{\widehat{GVT}-1} \sum_{d_k=1}^{j_k}\right) \sum_{d_n=1}^{j_n} [d_n \times f_{N_{max(j_0-c,0)}}(i_0) \times rb(I_0,J_0) \times \prod_{k=1}^{n} \left(f_{N_{max(j_k-c,0)}(j_{k-1})} \times halt_{J_{k-1}}^{I_{k-1}}(d_{k-1}) \times rb(I_k,J_k)\right) \times halt_{J_n}^{I_n}(d_n)]$$

The average cascading rollback distance caused by the $n$-th wave of negative messages is

$$\overline{D_n^-} = \sum_{j_n=0}^{\widehat{GVT}-1} \frac{D(j_n+a)_n^-}{\widehat{GVT}}.$$

## 4.4 Elapsed Time Characterization

The elapsed time of TW scheme is modeled by *computation time* and *communication time*. Let $N_{Total}$ be the total number of (true and false) events executed, $N_{Arr}$ the total number of (true and false) arrival events, and $N_{Dep}$ the total number of (true and false) departure events.

As the execution of each true event has expected rollback distances $\overline{D^+}$, $\overline{D_1^-}$, $\overline{D_2^-}$, ..., $\overline{D_l^-}$, where $l$ is the last wave. We have

$$\begin{aligned} N_{Total} &= N_p \times (1 + \overline{D^+} + \sum_{n=1}^{l} \overline{D_n^-}) \\ N_{Arr} &= \begin{cases} \frac{N_{Total}}{2} & \text{if } \lambda < \mu \\ N_{Total} \times \frac{\lambda}{\lambda+\mu} & \text{otherwise} \end{cases} \\ N_{Dep} &= N_{Total} - N_{Arr}. \end{aligned}$$

The cost of processing each event includes the event execution time and state-saving time. We have

$$T_{Comp} = N_{Total} \times (T_{event} + T_{state}).$$

Based on the communication time accounted in figure 2, we have

$$T_{Comm} = N_{Arr} \times T_{buffer} + N_{Dep} \times (T_{buffer} + T_{transit}).$$

The elapsed time of TW simulator is given as follows:

$$\begin{aligned} T_{TW} &= T_{Comp} + T_{Comm} \\ &= N_{Total} \times (T_{event} + T_{state}) + \\ &\quad N_{Arr} \times T_{buffer} + N_{Dep} \times (T_{buffer} + T_{transit}) \end{aligned}$$

### 4.5 Time Warp Efficiency

Since the TW simulator may be trapped in a racing state, it is also important to analyze the efficiency ($\mathcal{E}$) of the mechanism. We define the TW efficiency as the percentage of the committed events, i.e.,

$$\begin{aligned}\mathcal{E} &= \frac{N_p}{N_{Total}} \times 100\,\% \\ &= \frac{1}{1 + \overline{D^+} + \sum_{n=1}^{l} \overline{D_n^-}} \times 100\,\%\end{aligned}$$

## 5 Model Validation and Performance Analysis

We implemented an optimistic MIN simulation model (figure 6) in C language on a Fujitsu AP3000 parallel computer. Two versions of optimistic scheme are implemented: conventional TW and throttled TW. PVM software was used for spawning the simulation processes and for handling message passing. The simulation model results presented are based on simulating a 8 x 8 Omega MIN, with $\lambda = \mu = 100, \beta = 200$, $GVT = 20$ and $r = 0.75$. A heuristic based on opportunity cost, that derives $r = 0.75$ for close-to-optimal performance, is given in [16].

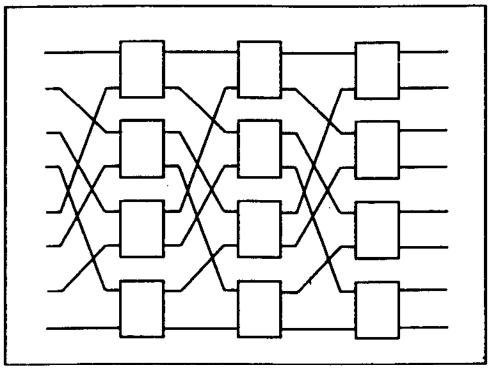

Figure 6: Multistage Interconnection Network

Four parameter values used in the analytical model, namely $T_{event} = 1000$, $T_{state} = 800$, $T_{buffer} = 2500$, $T_{transit} = 2000$, are obtained by taking measurements on a Fujitsu AP3000 distributed-memory parallel computer. In cases where the parallel simulator has not been implemented and the computer is not available, we may estimate the parameters from the existing sequential or similar parallel simulators and machine specifications. Due to space constraint, we present the performance of throttled TW in this paper. The performance for the unthrottled TW is given in [15].

### 5.1 Stability and Accuracy

We first analyze the performance model stability. Figure 7 shows that the predicted rollback distances grow gradually during the warming-up period and become stable. In addition, the cascading rollback distance becomes stable earlier (at $a \approx 10$) than that due to straggler (at $a \approx 25$). As such, while predicting the TW performance we need not analyze the LVT advancement for the whole duration of simulation run. In fact such an analysis is not possible due to the numerical overflow problem for large numbers, and the huge amount of computing time required. Instead we only need to analyze the progression delimited by a stable GVT window, and then expand the result for the whole duration. In the following validation we let $a = 30$ for the lower bound of GVT window. The communication delay is $c = \lceil\frac{2\times T_{buffer}+T_{transit}}{T_{event}+T_{state}}\rceil = \lceil\frac{2\times 2500+2000}{1000+800}\rceil = 4$ (events). We let the standard deviation of the normalized discrete distribution as $\chi = c$.

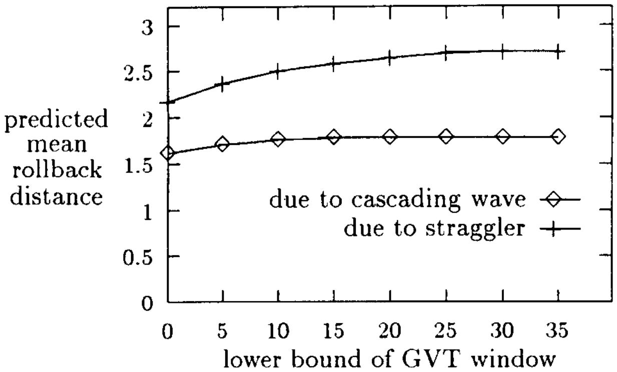

Figure 7: Model Stability with Respect to the Advancement of GVT Window

### 5.2 Sensitivity Analysis of the Throttled Time Warp Scheme

Figure 8 shows the efficiency of throttled TW when the spread of LP states is varied. As observed, the TW efficiency is improved if $\chi$ is decreased, i.e., as the state indices of sender and receiver become closer to each other, the number of rollback events is also reduced. This spread reduction can be achieved by augmenting a parallelism throttle in TW to maintain a more in-pace LVT advancement during the simulation run. Figure 9 shows the elapsed time of throttled TW scheme. The measured elapsed time for throttled TW simulation is stable due to the control of individual LVT progression, thus reducing the number of rollback occurrences. The proposed performance model also shows close elapsed time predictions for throttled TW simulation.

## 6 Conclusions

We proposed an analytical model that includes state saving cost and communication delay. The model provides a practical framework to analyze performance of TW even before the simulator is implemented. The input data required can be either estimated or obtained

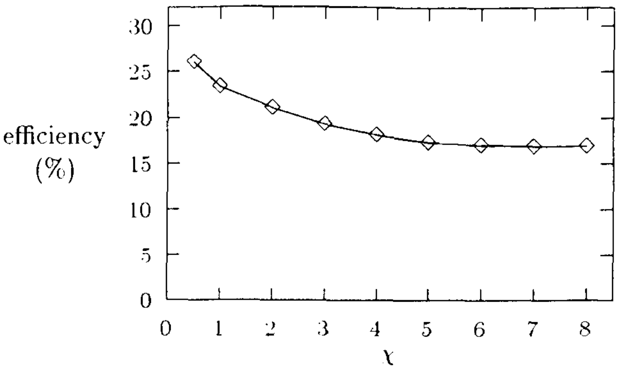

Figure 8: Efficiency of Throttled TW depending on the Spread of LP States

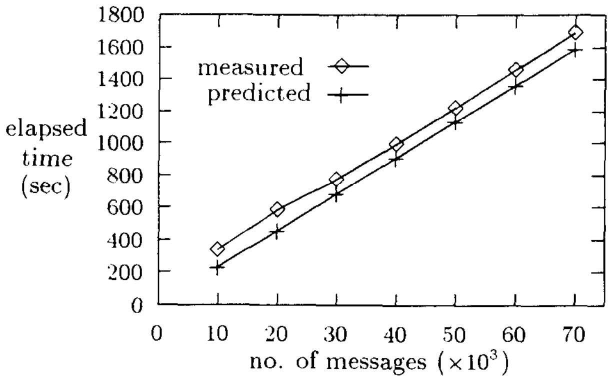

Figure 9: Elapsed Time for MIN Simulation (Throttled)

from a sequential simulator (or similar parallel simulators), and hardware specifications. Performance measures are derived for rollback probability and rollback distance, simulator elapsed time and efficiency. Validation experiments comparing the performance metrics from the analytical model and the simulation model show 10% difference. Rendering the difference of LP state numbers mimics the degree of TW event speculation, thus the proposed model can be used to analyze throttled TW simulation. The analytical framework can also be extended to analyze heterogeneous simulation and platform by using different parameterizations of LVT advancement rates and communication delays respectively. More importantly, our proposed model has encapsulated the effect of communication delay and cascading rollback, which are the main factors causing performance degradation in many TW simulators.

## References

[1] I. F. Akyildiz, L. Chen, S. R. Das, R. M. Fujimoto and R. F. Serfozo, *"The Effect of Memory capacity on Time Warp Performance,"* Journal of Parallel and Distributed Computing Vol. 18, pp. 411-422, 1993.

[2] P. M. Dickens, D. M. Nicol, P. F. Reynolds Jr. and J. M. Duva, *"Analysis of Bounded Time Warp and Comparison with YAWNS,"* ACM Transactions on Modeling and Computer Simulation,, Vol. 6 (4), pp. 297-320, October 1996.

[3] R. E. Felderman and L Kleinrock, *"Bounds and Approximations for Self-Initiating Distributed Simulation Without Lookahead,"* ACM Transactions on Modeling and Computer Simulation, Vol. 1(4), pp. 386 - 406, 1991.

[4] A. Gupta, I. F. Akyildiz, R. M. Richard, *"Performance Analysis of Time Warp With Multiple Homogeneous Processors,"* IEEE Transactions on Software Engineering, Vol. 17 (10), pp. 1013 - 1027, 1991.

[5] D. R. Jefferson, *"Virtual Time,"* ACM Transactions on Programming Languages and Systems, Vol. 7 (3), pp. 404-425, July 1985.

[6] L. Kleinrock, *"On Distributed Systems Performance,"* Computer Networks ISDN Journal, Vol.200 (1-5), pp. 209-216, 1990.

[7] S. Lavenberg, R. Muntz and B Samadi, *"Performance Analysis of a Rollback Method for Distributed Simulation,"* Performance' 83. Amsterdam: North-Holland, pp. 117-132, 1983.

[8] Y. B. Lin and E. Lazowska, *"Optimality Consideration for Time Warp Parallel Simulation,"* In Proc. of 1990 SCS Multiconference on Distributed Simulation, pp. 29-34, 1990.

[9] B. Lubachevsky, A. Weiss and A. Shwartz, *"An Analysis of Rollback-Based Simulation,"* ACM Transactions on Modeling and Computer Simulation, Vol. 1 (2) pp.154-193, 1991.

[10] D. Mitra and I. Mitrani, *"Analysis and Optimum Performance of Two-Message Passing Parallel Processors,"* Synchronised by Rollback, Performance Evaluation Journal, Vol. 7, pp. 111-124, 1987.

[11] D. M. Nicol, *"Performance Bounds on Parallel Self-Initiating Discrete-Event Simulation,"* ACM transaction on Modeling and Computer Simulations, Vol. 1(1), pp 24-50, 1991.

[12] B. D. Plateau and S. K. Tripathi, *"Performance Analysis of Synchronization for Two Communicating Processes,"* Performance Evaluation Journal, Vol. 8, pp. 305-320, 1988.

[13] J. Steinman, *"SPEEDES: A Multiple-Synchronization Environment for Parallel Discrete-Event Simulation,"* International Journal in Computer Simulation, Vol. 2, pp. 251-286, 1992.

[14] S.C. Tay, Y.M. Teo and S.T. Kong, *"Speculative Parallel Simulation with an Adaptive Throttle Scheme"*, Proc. of 11th Workshop on Parallel and Distributed Simulation (PADS'97), Lockenhaus, Austria, IEEE Computer Society Press, pp. 116-123, June 10-13, 1997.

[15] S.C. Tay, Y.M. Teo and R. Ayani, *"Performance Prediction of Optimistic Simulation with Rollback Thrashing"*, Technical Report TR41/97, Department of Information Systems and Computer Science, National University of Singapore, pp. 1-24, November 1997.

[16] S.C. Tay, *"Parallel Simulation Algorithms and Performance Analysis"*, Ph.D. Thesis, National University of Singapore, Dept. of Information Systems and Computer Science, 1998.