Simulations of the implementation of primary copy two-phase locking in distributed database systems

S Vasileva*

College - Dobrich, University of Shumen, Dobrich, Bulgaria

**Corresponding author's e-mail: svetlanaeli@dobrich.net*

Received 1 March 2015, www.cmnt.lv

Abstract

This paper considers algorithms for concurrency control in Distributed database (DDB) systems. Below are the simulating models of the implementation of two-phase locking (2PL) in DDB. From four types 2PL in DDB (Centralized 2PL, Primary copy 2PL, Distributed 2PL and voting 2PL) is viewed Primary copy 2PL, as this protocol is a "transitional" protocol of Centralized 2PL to the Distributed 2PL. The paper describes specifically the simulations of two-version 2PL and 2PL with integrated timestamp ordering mechanism. In concurrency control method 2PL may take place deadlocks of the transactions. Therefore, in the modelling algorithms described here are integrated algorithms for deadlock avoiding: two-version architecture of database and timestamp ordering strategy "wait-die". There are also presented, the results of the simulations of these two variants of the 2PL method at different scales of the networks for the transmission of data and at different intensities of inflow transactions. Modelling algorithms are developed by means of the system for simulation modelling GPSS World Personal Version.

Keywords: Simulation models, concurrency control, distributed transactions, 2PL, distributed database

1 Introduction

Concurrency control techniques are generally divided into: Locking, Timestamp ordering and Optimistic strategies - Validation check up. [1] In the last two methods were obtained a better retention of transactions in the system when it is saturated with conflicts (due to frequent rollbacks of transactions). Therefore, it is desirable to use the method of Two-phase locking (2PL). But emerging problems in its application require testing. One effective and inexpensive method for testing the operation of the various systems is the method of simulation. The basis of design are used presented and examined in [2] and [3] simulation algorithms, primary copy two-version 2PL and primary copy 2PL with built mechanism of Timestamping (TS) strategy "wait-die". In the modelling algorithms the data are subjected to incomplete replication (all data items have the same number of copies; the number of copies of the data item is smaller than the number of sites). The models were developed by means of the "classical" system modelling - GPSS World.

Since pessimistic protocols can arise deadlocks transactions, the problems arise to detect and resolve deadlocks. One way to avoid deadlocks is the use of data architecture with many versions in Distributed Database Management Systems (DDBMS). In [4] are presented and discussed algorithms with many versions for concurrency control in database management systems (DBMS), and in particular: Multi-version 2PL and Two-version 2PL. Since in the Multi-version 2PL there are problems with the management and conservation of the versions, to benefit of its advantages is desirable to limit the number of versions. So look Two-version Two-phase locking (2V2PL).

Besides algorithms with versions, were developed and protocols for concurrency control, which combine advantages of 2PL and Timestamping method. In the paper we consider the model of such an algorithm - Model of Primary copy 2PL with integrated mechanism of Timestamping - strategy "wait-die" (This method is described in [5] and some others).

2 Primary Copy 2V2PL Model

A protocol with two versions for two-phase locking is first proposed by Bayer in [6]. In 2V2PL protocol are only two versions of elements: 1 current version of the item and not more than one incomplete version. We use 3 types of locks, each lock is released at the end of the transaction: rl - (read lock) - established on the current version of the data item shortly before reading it; wl - (write lock) - sets before creating the new (incomplete) version of the item; cl - (commit lock) - established before the implementation of the last transaction of the transaction (usually before surgery commit) on each data item that it has recorded.

Advantages of the protocol: limit the number of older versions leads to reduction of volume of database (DB); does not require special storage structure versions; simple enough to implement a protocol.

Disadvantage: An organization of versions is advantageous especially for transactions that operation "reading". Problem: Not solved effectively the issue of transactions executed operation "record".

With the development and implementation of algorithms with many versions (MultiVersion Concurrency Control - MVCC) to solve many of the problems in algorithms with one version for concurrency control, but not all developed algorithms with versions are actively used in practice due to the problems of creating a DBMS with versions, namely: - Necessity of the construction of physical data organization, providing effective management of versions - Reporting: the expansion of the volume of DB adding versions; possible restrictions on the number of versions, but in practice such a restriction is often lacking - the system simply delete old versions; - Need for reporting the versions in other components of the DBMS. In practice, the most frequently are used the protocols Read-Only MultiVersion and 2V2PL, because they, according to [4] are simple enough to implement and provide the user the advantages of most slurry conversion DBMS. In 2V2PL protocol it is solved the problem with the limitation of the number of versions.

The presented simulation models use 6 generated streams of transactions which imitate global transactions in DDB systems. They are all in parallel streams and their intensity λ is given in tr per sec (number of transactions per second). The structural scheme of a modelling algorithm for Primary copy 2PL of distributed transactions management in two-version architecture of data in Distributed DB (DDB) is shown in Figure 1.

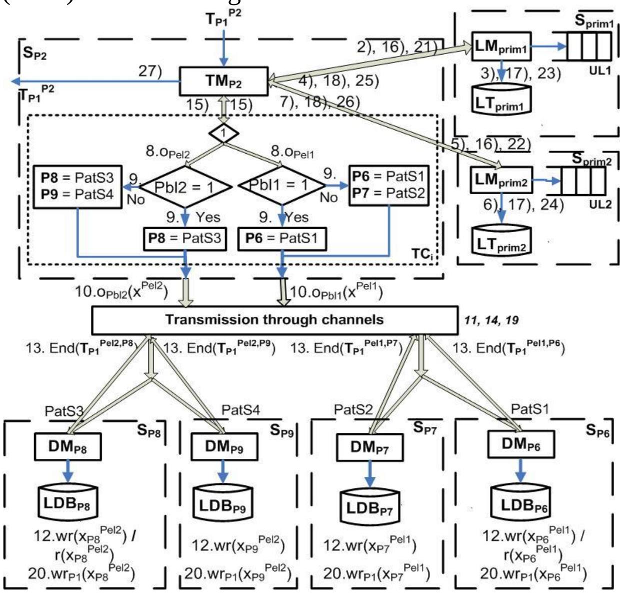

FIGURE 1 Scheme of Primary copy 2V2PL

The basic steps in the synthesized Primary Copy 2V2PL algorithm [2] are:

1. Transaction coordinator TC_{P2} splits global transaction T_{P1}^{P2} in two sub-transactions, T_{P1}^{Pel1} and T_{P1}^{Pel2}, for the process of the two elements *El1* and *El2* (operation 1 on Figure 1). These sub-transactions T_{P1}^{Pel1} (operation 2 on Figure 2) and T_{P1}^{Pel2} (operation 5 on Figure 1) are directed at lock managers LM_{prim1} and LM_{prim2}, to get the proper locks for the elements *El1* and *El2*.
2. If it is allowed for sub-transactions to lock the elements *El1* and *El2*, then the correspondent records are put in the lock tables LT_{prim1} and LT_{prim2} (operations 3 and 6 on Figure 1). Moreover the sub-transaction T_{P1}^{Pel2} waits for the acknowledgment of getting the locks of *El1* by T_{P1}^{Pel1} in S_{P2}. After that it is directed to the lock manager LM_{prim2} (operation 5 on Figure 1).
3. After given and granted locks (operations 4 and 7 on Figure 1), the transaction T_{P1}^{Pel1} splits in two sub-transactions $T_{P1}^{Pel1,P6}$ and $T_{P1}^{Pel1,P7}$ (operations 8 and 9). If the transaction T_{P1}^{Pel1} is required to read only the element *El1* then it doesn't split in two separate sub-transactions. Similarly, the transaction T_{P1}^{Pel2} splits in two-transactions $T_{P1}^{Pel2,P8}$ and $T_{P1}^{Pel2,P9}$ depending on where the copies of the data elements processed by the transaction T_{P1}^{P2} have been stored.
4. Sub-transactions $T_{P1}^{Pel1,P6}$ (and $T_{P1}^{Pel1,P7}$) and $T_{P1}^{Pel2,P8}$ (and $T_{P1}^{Pel2,P9}$) are directed to the corresponding data managers (operations 10 on Figure 1). After that they are transmitted in the network (operations 11 on Figure 1) to DM_{P6} (and DM_{P7}) and DM_{P8} (and DM_{P9}) If they have to be committed in a remote node.
5. Many operations read/write are committed in the sites-executors, where the corresponding data managers are included. The nodes PatS1 and PatS3 perform operations read/write over the first replica of *El1* and *El2*. It is shown in fig.1. The nodes PatS2 and PatS4 perform the operations write second replica of *El1* and *El2* (operations 12 on Figure 1 and Figure 2).
6. If the sites-executors don't match to the site – initiator of T_{P1}^{P2} after the sub-transactions have finished their actions in sites S_{P6}, S_{P7} and S_{P8}, S_{P9}, they are transmitted over the communication network to the transaction manager (operations 13 and 14 in Figure 1).
7. Transaction manager of T_{P1}^{P2} sends messages, which consists of requests for releasing the elements *El1* and *El2* (operations 16) to the correspondent lock manager, when it receives messages for ends of sub-transactions (operations 15 on Figure 1).
8. If a new version of El1 and/or El2 is created by the transaction then the requests for end of record and release of exclusive lock wl (operations 16 on Figure 1) are getting transformed to requests for certify lock cl. Moreover higher priority is given to them. There is existed a situation, in which the correspondent lock manager permits a record of certify lock of *El1* and/or *El2* (operations 17 on Figure 1) to be included in lock table, when acknowledge of certify lock (operations 18 on Figure 1) is received. In this common case, the sub-transactions T_{P1}^{Pel1} and/or T_{P1}^{Pel2} are sent through the network toward to the sites – executors to make uncommitted version of *El1* and/or *El2* a current one.
9. After fixing the results from sub-transaction processing in sites S_{P6}, S_{P7} and S_{P8}, S_{P9} $T_{P1}^{Pel1,P6}$ and/or $T_{P1}^{Pel1,P7}$ and $T_{P1}^{Pel2,P8}$ and/or $T_{P1}^{Pel2,P9}$, the transactions are sent through the communication network to the transaction manager (operations 20 on Figure 1).
10. Transaction manager of T_{P1}^{P2} sends messages to the corresponding lock manager when it gets the messages for the end of the sub-transactions. These messages contain requests to request a release for locks of the elements *El1* and *El2* (operations 21 and 22).
11. Sub-transactions T_{P1}^{Pel1} and T_{P1}^{Pel2} release locks of *El1* and *El2*. The records for these elements have been deleted from the lock tables (operations 23 and 24 on Figure 1). Afterwards, acknowledgments for the releasing of locks for *El1* and *El2* are sent to the manager of T_{P1}^{P2} (operations 25 and 26 on Figure 1).
12. Transaction T_{P1}^{P2} quits the system (operation 27 on

Figure 1) as soon as sub-transactions T_{P1}^{Pel1} and T_{P1}^{Pel2} finish their process.

3 Model of Primary copy 2PL with TS "wait-die"

The method uses the so called timestamps (TS) of the transactions. There are two possible algorithms when transaction Ti wants to receive a lock of an element [5]:

1. "wait – die": If T_i is older (with smaller value of timestamp) than T_j, which has blocked the element then T_i waits for the release of the element in order to put the lock. If T_i has a greater timestamp than T_j, then T_i restarts.
2. "wound – wait": If T_i is younger (with greater timestamp value) than T_j, which has blocked the element, then T_i waits for the release of the element in order to put the lock. If T_i has a smaller timestamp than T_j, then T_i restarts.

In the Timestamp method the number of "superfluous" rollbacks is much smaller. Moreover the method is not very difficult to be realized, especially variant 1 „wait – cancel". Therefore we view one such realization of the Timestamp method in embedding in Primary copy 2PL algorithm for DDBMS.

The basic steps in the synthesized Primary Copy 2PL with TS model [3] are:

1. When the transaction T_{P1}^{P2} comes in the transaction manager TM_{P2} its length is checked (1 or 2 data elements will be processed) - operation 1 on Figure 1 and the transaction is prepared to be split (operations 8 on Fig. 2).
2. With the operations 9 values of the parameters of the sub-transactions are acquired – the numbers of the data managers DM_{P6}, (DM_{P7}), (DM_{P8} and DM_{P9}), where the sub-transactions $T_{P1}^{Pel1,P6}$, ($T_{P1}^{Pel1,P7}$), ($T_{P1}^{Pel2,P8}$ and $T_{P1}^{Pel2,P9}$) have to execute the operations of reading/recording of the copies of data elements *El1* and *El2*.
3. After the primary processing in the transaction coordinator TC_{P2} the requests for locking *El1* and *El2* are transmitted through the net to the corresponding primary lock managers LM_{prim1} and LM_{prim2} (operations 2 and 5).
4. LM_{prim1} and LM_{prim2} check in the lock tables LT_{prim1} and LT_{prim2} if the lock of El1 and El2 is allowed (operations 3 and 6 on Figure 2). If the lock of *El1* (and *El2*) is allowed, the corresponding record is put opposite the number of the element in LT_{prim1} (and LT_{prim2}).
5. The transaction receives confirmation messages about the lock of *El1* (operation 4) and if two data elements are being processed, TM_{P2}, through the transaction coordinator TC_{P2} sends the request for lock of *El2* to LM_{prim2} (operation 5 on Figure 2).
6. If the lock of the corresponding element is not possible, the number of the transaction is check if it is smaller than the number of the transaction which has put the lock:
   a. if the sub-transaction is not going to continue and is not going to restart, it waits the release of the element in user chain, whose number is the number of the element;
   b. if the sub-transaction has not received the lock of the element it restarts (operation 4/operation 7 is a restart operation). After it has arrived in TM_{P2}, the restarted lock request (operation 16 on Figure 2) is transmitted to LM_{prim1}/LM_{prim2} (the repeated (successful) attempt for lock element 1/element 2 is presented with operations 16, 17 and 18).
7. Transaction which has finished with the operation read/write releases the element in LT_{prim1} (and LT_{prim2}) – operations 21 and 22 on Figure 2. The requests for release of the lock of the elements are transmitted to the corresponding primary lock manager with operations 19 and 20.
8. After the release of the lock of an element, the transaction which is first in the waiting list heads to the lock manager. If it is a group of sub-transactions then they receive a shared lock of the element.
9. Receiving a confirmation for a lock of the elements of the GPSS transaction being allowed, a modelling global transaction splits. After that the sub-transactions are transmitted through the net to the data managers for executing the read/write operations (operations 10 and 11 on Figure 2).
10. The sub-transactions of T_{P1}^{P2} execute read/write in local databases LDB_{P6}, LDB_{P7}, LDB_{P8} and LDB_{P9} with the corresponding replicas of *El1* and *El2* (operations 12 on Figure 2). After that they are transmitted to the transaction manager TM_{P2} (operations 13 and 14). If a transaction renews a data element, the sub-transactions recording the corresponding copies wait for each other and get united (operations 15 on Figure 2), before a request for release of the lock of the element is sent to LM_{prim1} (and LM_{prim2}).
11. Transaction T_{P1}^{P2} quits the system (operation 25 on Figure 2) as soon as sub-transactions T_{P1}^{Pel1} and T_{P1}^{Pel2} finish their process (modelled with operations 23 and 24).
12. The transfer through the network to primary lock managers LM_{prim1} and LM_{prim2} and to the sites-executors, where are the data managers DM (operations 2, 4, 11, 14, 16, 18, 19, 21, 25 on Figure 1 and operations 2, 4, 11, 14, 16, 18, 19, 23 on Figure 2) is simulated with retention given by the matrix *MX$RAZST* for both models.

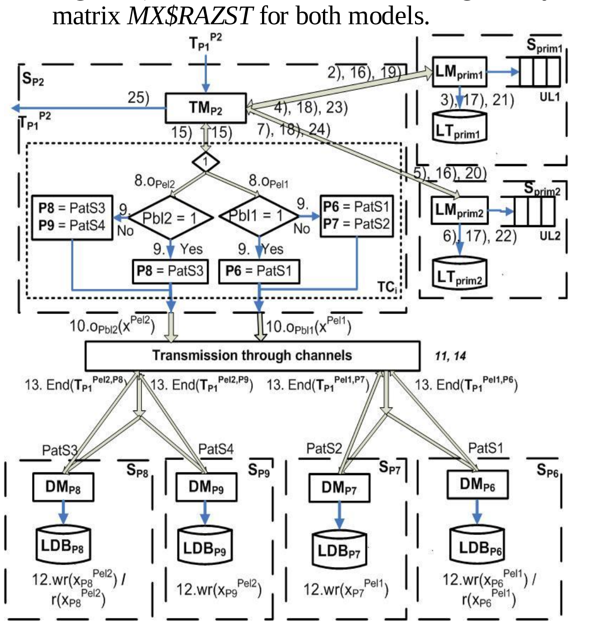

FIGURE 2 Scheme of Primary copy 2PL with TS "wait-die"

4 Matrices used in the modeling algorithms

In the synthesized simulation models mainly is used the matrix MX$LTA for modeling the lock table. Each row of the table corresponds to the data element from DDB. The matrix has the following columns:

- Type of the resulting lock the GPSS transaction blocked free data elements. This column, records the value parameter P$bl1 or P$bl2 of the same (depending on whether the element is the first or the second processed by transaction);
- Number the GPSS transaction, blocked free element. This column records the value the P1 parameter transaction, which placed the lock in the first column;
- Number of the site - initiator the GPSS transaction blocked first element (the value of parameter P2 of the transaction, borrowed the element in the lock table);
- Number of the GPSS transaction, received a shared lock element, blocked by another "reading" transaction before. This column records the value the P1 parameter of the transaction, whose request for "reading" is compatible with the shared lock set by the first column;
- Number of the GPSS transaction, received an interlock "record" elements blocked by "reading" transaction before. This column records the value the P1 parameter of the transaction, whose request for "record" is compatible with the shared lock, set by the first column 2V2PL modeling algorithm.

In the synthesized simulation models are used also and two matrices - *MX$RAZST* and *MX$RAZDEV* to set the mean and standard deviation of the retention time of he transactions in the transmission of messages between the nodes of the distributed database system modeled for communication costs.

5 Auxiliary variables and cells

In the simulation models primarily are used the following variables and cells serving as counters:

V$ElemN1 – in it are calculated random selection numbers of the first element that will process the transaction;

V$ElemN2 – in it are calculated the random selection of the numbers of the second element which will process the transaction. In the statements which calculates variable ElemN1 and the ElemN1 is involved and the random number generator RN2;

V$RAZRBL1/V$RAZRBL2 – in it is calculated the admissibility of the first/second lock element processed by the transaction. The value is the product of the parameter P$bl1/P$bl2 of the GPSS transaction requesting the lock and the value in the first column of the matrix LTA;

X$BROITR – total number of generated GPSS transactions during the modeling;

X$ZAVTR – total number of transactions leaving the model served.

6 Parameters of the GPSS transactions

P1 – Number of transaction. The value is a sum of System Numeric Attribute MP2 (The subtraction between the relative model time and the content of the second parameter of GPSS transaction) and the number of the site;

P2 – Number of the site, where the transaction is generated. The value is a number from 1 to <number of stream transactions>;

P$Nel – Length of the modeled transaction. The value of that parameter in the constructed models is 1 or 2 chosen by probability defined by the function FN$BrEl respectively 0.30 and 0.70. It is supposed that long transactions get in the system more frequently then short ones in that model;

P$El1/P$El2 – Number of the first/second element, which the generated transaction will read or write. The value is a random number and is uniformly distributed in the interval [1, NumEl];

P$bl1/P$bl2 – Type of the requested lock for the first/second element, which will be processed by the generated transaction;

P5 – Value 0, if the transaction is in 1st phase – occupation of the locks and value 1, if the transaction finishes its work and has to release the locks. In the future value 2 will be able to be appropriated to a transaction which will have to be restarted;

P$Prim1/P$Prim2 – Number of the primary site of the first/second element, which the generated transaction will read or write;

P6 and P7 – In them there are correspondingly recorded the number of the site, where it is the nearest copy of the data element and the number of the site, where it is the second replica of the first data element, processed by transaction. Correspondingly in parameters P8 and P9 we have the nearest copy of the second data element and the number of the recorded site, where it is the second replica of the second data element.

P$vr – This parameter is used in the model of 2PL with timestamping - P$vr = 1 in the case that the lock of the element is not possible and the number of the GPSS subtransaction is smaller than the number of the transaction, which set the lock, then, according to the timestamping mechanism strategy "wait-die" [5] if the subtransaction does not continue and does not restart, it waits for the release of the elements in the user chain, whose number is the number of elements.

7 Simulation Results

Our researches has been made for 2 replicas of the element, number of the incoming streams - 6 and number of the data elements in the global database - 50. Several parameters have been changed in order to obtain the whole picture of the states in which the chosen models went through.

The parameters and indexes of the simulations of the considered model are as follows: NumTr – general number of the generated transactions for the time of incoming modelling; FixTr – general number of the completed (committed) transactions for the same period; X=FixTr/Tn – throughput of the queuing system; Tn – time interval in which the system is being watched; Ps=FixTr/NumTr – probabilities for transaction service. The results are received in 6 streams of concurrent transactions with different intensity. The copies of the data elements are distributed evenly and random by 6 sites in the system. Service and rejection probabilities are calculated according to closely associated formulas and the received values are different from those received through more detailed expressions.

Time modelling developed in the GPSS World algorithms is set in milliseconds. All streams transactions are received upon an exponential law with a variable at different studies with an average length of the interval. In all modelling algorithms we consider 6 streams generated by GPSS transactions modelling 6 sites in distributed database system, from which Poisson law shall go global transactions.

The diagram of Figure 3 presents the results of simulations of Throughput of Primary copy 2PL algorithms at the same intensities of input flows depending on the monitoring period (in seconds): The graph marked with a thin blue dashed line (2V2PL) and the graph indicated by a thick black line and square markers (2PL TSwd) – 6 streams, each with an average intensity 4,17 tr/s (minimum load - intensity cumulative flow 25 tr/s); The graph marked with thin black line and asterisks (2V2PL) and in the graph illustrated by dashed lines with triangular markers (2PL TSwd) - 6 streams, each with an intensity of 8,33 tr/s (average load - intensity of the aggregate stream 50 tr/s); The graph indicated by the thin dotted line (2V2PL) and the graph indicated by a thin blue line (2PL TSwd) - 6 streams of medium intensity 16,67 tr/s (maximum load - intensity of aggregate stream 100 tr/s).

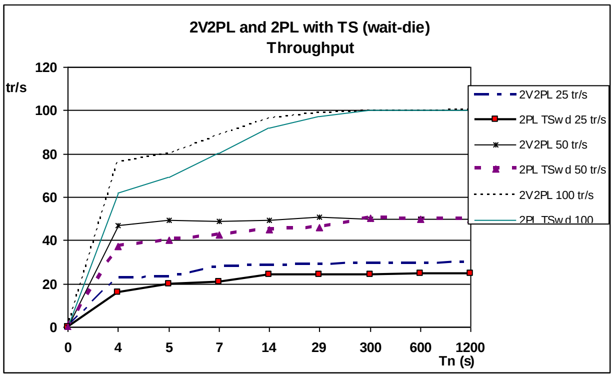

FIGURE 3 Throughput of the systems

The diagram in Figure 3 shows the Throughput for both models at three operating modes of load increased with the increase of the model during and after a certain time (5 minutes after the start of operation of the system); the occurrence of stationary mode reaches a maximum value and does not change with time.

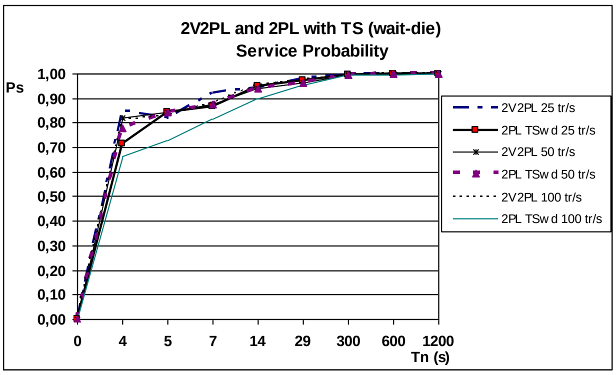

FIGURE 4 Service probabilities in the 2V2PL model and the 2PL with TS "wait-die" model

Service probability factor or completeness of service transactions serves to assess the dynamic properties of DDBMS. Figure 4 presents the results for the probability of service of distributed transactions at simulation algorithm Primary copy 2V2PL and Primary copy 2PL with an integrated mechanism of timestamping at the same intensities of inflows (as for Figure 3).

On Figure 5 graphically are given the values that are obtained for throughput by substituting the fixed in the receiving reports values of the cell X$ZAVTR. Intensity of inflows transactions are the same as the graphs of fig. 3, was changed only the matrix of distances MX$RAZST - the values in it are increased twice compared to models whose results are reported in the graphs of Figure 3.

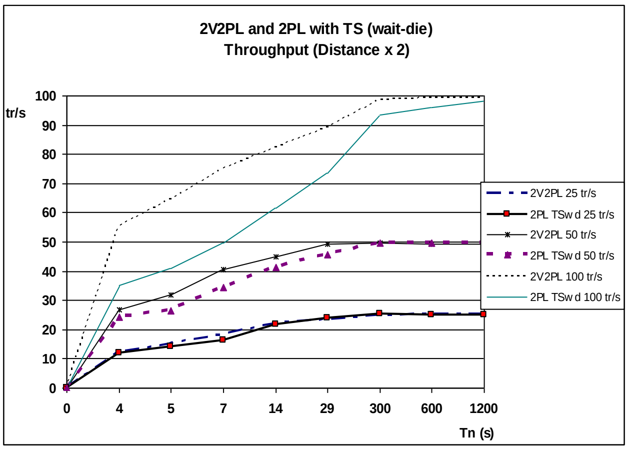

FIGURE 5 Throughput in the models at doubled distance between the sites in the system

From the graphs of Figure 5 it can be concluded that with the increase of the distance between sites in the system, the throughput graphs are "spaced apart" more. This is very evident in the graphs at maximum load of the system (intensity of inflow 100 tr/s).

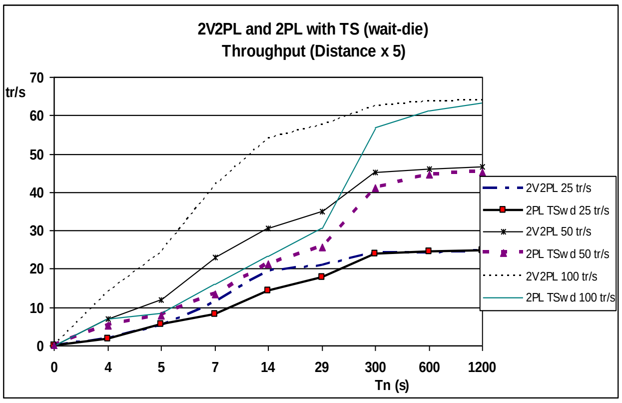

FIGURE 6 Throughput of the system (distance x 5)

By varying the values of cells in the matrix MX$RAZST (in the GPSS models) we can conduct research whether and how the gaps in the net, impact on Throughput, Service probability and other performance indicators of concurrency control algorithms.

For example, by five times, increasing the cell values of the matrix MX$RAZST are obtained the values of the throughput graphs on Figure 6 from the simulation reports of the Primary copy 2V2PL protocol and the Primary copy

2PL with timestamping "wait-die" at the same intensities of inflows as Figure 3.

By increasing the distance between sites in the system, except the increasing divergence of the graphs throughput, is observed slower occurrence of stationary mode. This is shown in the graph for the Primary copy 2PL with timestamping maximum intensity of the inflow transactions - graph, presented by blue line (*2PL TSwd 100 tr/s*) in Figure 7.

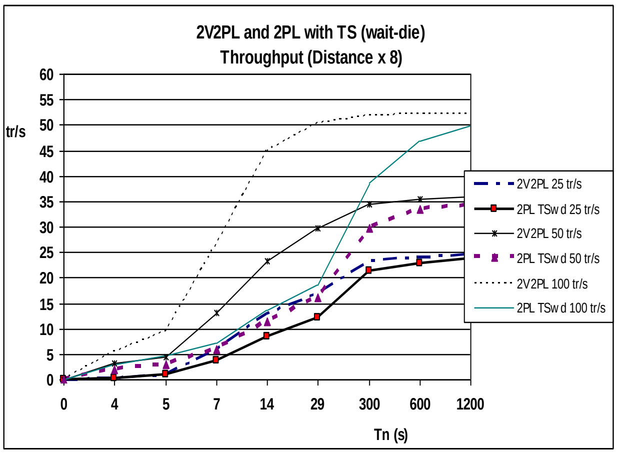

FIGURE 7 Throughput of the system (distance x 8)

Another indicator of the performance of concurrency control algorithms is Frequency distribution of Response time (RT) of transactions. Diagrams of Frequency distribution of RT are built automatically by the formulated in the GPSS model tables. On Figure 8 is demonstrated the histogram of Frequency distribution of RT in modeling Primary copy 2V2PL at the total intensity of the input streams 100 tr/s (maximum load on the system) and observation time 28.8 seconds. A histogram is generated automatically by GPSS World on table Response time distribution DATABLE, whose values are brought into the simulation report.

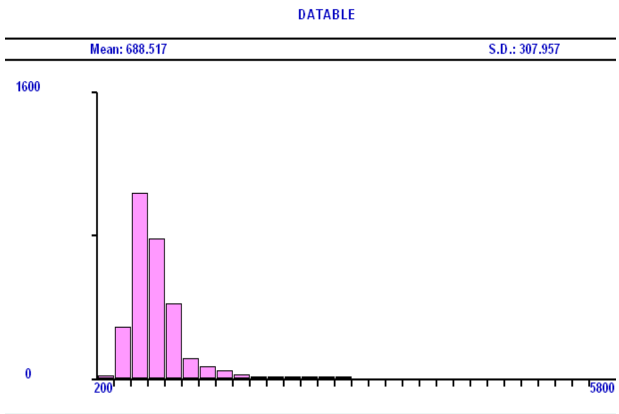

FIGURE 8 Frequency Distribution of RT in modelling Primary copy 2V2PL at λ = 100 tr/s

On fig. 9 is demonstrated the histogram of Frequency distribution of RT in modeling Centralized 2PL with timestamping (wait-die) at total intensity of the input streams 100 tr/s (maximum load on the system) and observation time 28,8 seconds (as in Figure 8).

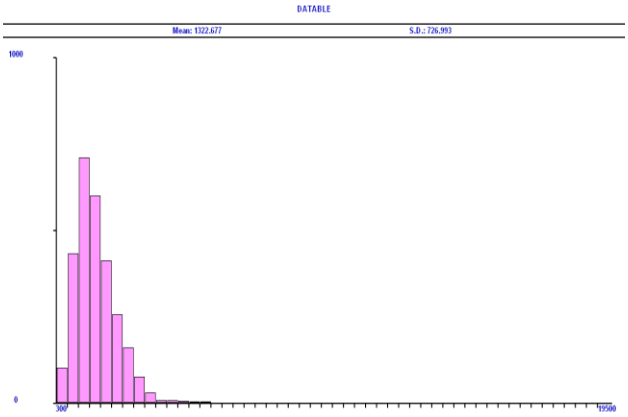

FIGURE 9 Frequency Distribution of RT in modelling Primary copy 2PL with TS (wait-die) at λ = 100 tr/s

The tables of Frequency distribution of RT besides that serve comparative analysis of concurrency control algorithms, serve also to assess the reliability of modeling algorithms by comparing with the template chart of Frequency distribution of RT [7].

Similarly, can be compared with the template graphics and charts for throughput of Figures 3, 5, 6, 7.

This approach may be useful in developing an information system for mobile learning [8].

8 Conclusions

The system of simulation GPSS World permits creation of effective simulation models of transaction concurrency control (in particular models of Two-phase locking of transactions in DDBMS with one-version and two-version architecture of replicated data.

Conducted simulations and the results confirm functionality of the modelling algorithms.

Simulation models allow definition the throughput ability of the distributed systems, the average service time of distributed transactions and other parameters on the basis of which the efficiency of the suggested algorithms could be defined.

The comparative analysis of the results of modelling of one-version and two-version Primary copy 2PL showed no deadlocks, while 2V2PL throughput is high and the response time is significantly lower than that of the one-version 2PL algorithm.

Acknowledgments

This paper is supported by the Project BG051PO00l-3.3.06-0003 “Building and steady development of PhD students, post-PhD and young scientists in the areas of the natural, technical and mathematical sciences”. The Project is realized by the financial support of the Operative Program “Development of the human resources” of the European social found of the European Union.

References

[1] Srinisava R, Williams C, Reynolds P 2001 *Distributed transaction Processing on an Ordering Network* Technical Report CS-2001-08

[2] Vasileva S 2010 Modelling of Primary Copy Two-Version Two Phase Locking *IV International Conference on Information Systems and GRID Technologies* 79-91

[3] Vasileva S 2011 Algorithm for Primary Copy Locking with Timestamp Ordering V *International Conference on Information Systems and GRID Technologies* 236-47

[4] Chardin P 2005 Data multiversion and transaction concurrency control http://www.citforum.ru/database/articles/multiversion

[5] Connolly T, Begg C 2002 Database Systems Addison – Wesley 683

[6] Bayer R, Heller H, Reiser A 1980 Parallelism and Recovery in database systems *ACM Trans. Database Syst*. **5**(2) 139-56

[7] TPC Council 2010 "TPC – C Benchmark" Standard Specification Revision 5.11 Transaction Processing Performance Council (Feb)

[8] Nenkov N V, Ibryam I I Issa A 2011 Design and development of an information system for mobile learning using SQL and MYSQL Conference of National and World Economy *Application of ICT in the economy and education (ICAICTEE-2011)* 392-9

Author

Svetlana Vasileva, 1970, Plovdiv, Bulgaria

Current position, grades: Chief assistant professor, PhD in Informatics
University studies: Saint-Petersburg State Electrotechnical University (LETI); Computer Aided Design Systems
Scientific interest: Distributed database management systems; Simulation modeling of processes and systems; Application of information technologies in education
Publications: 80
Experience: a programmer (1995-1996 - "DataSoft" - Plovdiv); assistant professor of informatics (1997-2000 - Agricultural University - Plovdiv); Head computer lab (2000-2001 – Secondary School "Dimitar Talev" - Dobrich)