Analysis and Simulation of Networks with Virtual Cut-Through Routing

Lev B. Levitin
Boston University
levitin@bu.edu

Yelena Rykalova
University of Massachusetts, Lowell
yelena_rykalova@uml.edu

Abstract

This paper considers a model of a toroidal computer interconnection network with the virtual cut-through routing. The interrelationships between network parameters, load, and performance are analyzed. An exact analytical expression for the saturation point and expressions for the latency as a function of the message generation rate under the mean field theory approximation have been obtained. The theoretical results have been corroborated with the results of simulation experiments for various values of network parameters. The network behavior has been found not depending on the torus linear dimensions if they are at least twice as large as the message path length. The saturation point has been found to be inversely proportional to the message length in good agreement with the analytical results. A good agreement with Little's theorem has been found if the network remains in the steady state during the experiment.

1. Introduction

Modern approach to supercomputer design relies on massively parallel computers (MPCs) characterized by scalable architecture. As a result, these computers offer corresponding gains in performance as the number of processors is increased. Parallel code execution in such systems requires extensive communications between otherwise independent nodes. Since memory is not shared between node processors, interprocessor communications are achieved by passing messages between nodes through a communications network. This communications network is implemented as a network of interconnected routers, each having its local processor (see reference list). Many commercially available parallel computers use hypercube or mesh networks configuration. The same principles are observed in popular network-on-chip (NoC) architecture and routing techniques. These network configurations provide convenient modularization and required scalability.

Various routing techniques are used in interconnection networks (see Nikitin, & Cortadella, 2009; Kiasari, Lu, & Jantsch, 2013; Dabrowski, 2015; Li, Bashan, Buldyrev, Stanley, & Havlin, 2012; Chen et al., 2011; Liu & Carothers, 2011; Minkenberg, 2013; McCarthy, Isaacs, Bhatele, Bremer, & Hamann, 2014; Kermani & Kleinrock, 1979; Duato, Robles, Silla, & Beivide, 2001; Kaushal & Singh, 2014; Domkondwar & Chaudhari, 2012; Lekariy & Gaikwad, 2013; Wang, Ma, Lu, & Wang, 2014; Choudhary & Qureshi, 2012; Levitin,

Karpovsky, Mustafa, 2009; Levitin, Karpovsky, & Mustafa, 2006; Levitin, Karpovsky, & Mustafa, 2013; Karpovsky, Levitin, Mustafa, 2014; Sadawarte, Gaikwad, & Patrikar, 2011; Rexford & Shin, 1996; Hag, Hafizur Rahman, Nor, Sembok, Miura, & Inoguchi, 2015; Opoku Agyeman, Zong, Yakolev, Tong, & Mak, 2017; Daf & Saynkar, 2014). Store-and-forward approach is based on the assumption that an entire message must be received at any intermediate node before it can be forwarded to the next node. Obviously, for a long message, the total delivery time may turn out to be quite large.

To the contrary, in wormhole routing, each message is divided into small packets—flits. The header flit contains information about source and destination and is routed through the network according to this information and routing algorithm. Other flits of the message follow the header flit. When header flit of the message is blocked at an intermediate node because the requested link is occupied by another message, the flits are buffered at each node along the path up to the current node. This forms a long "worm" which remains in the network blocking other messages, thereby increasing their delivery time. Also, the problem of deadlocks emerges in this approach and should be dealt with (Levitin, Karpovsky, & Mustafa, 2009; Levitin, Karpovsky, & Mustafa, 2006; Levitin, Karpovsky, & Mustafa, 2013; Karpovsky, Levitin, & Mustafa, 2014).

Virtual cut-through (VCT) routing algorithm is supposed to mitigate the drawbacks of both the above-mentioned techniques (Chen et al., 2011; Minkenberg, 2013; Kermani & Kleinrock, 1979; Duato, Robles, Silla, & Beivide, 2001; Kaushal & Singh, 2014; Domkondwar & Chaudhari, 2012; Wang, Ma, Lu, & Wang, 2014; Sadawarte, Gaikwad & Patrikar, 2011; Rexford & Shin, 1996; Hag et al., 2015). Unlike the wormhole approach, in VCT routing, if the next node cannot accept the message, the current node must still be able to buffer the rest of the incoming message from previous nodes. Thus the VCT algorithm achieves a much higher throughput and avoids deadlocks at the expense of increased buffer capacity.

Several papers were devoted to the comparison between different routing techniques (Duato, Robles, Silla, & Beivide, 2001; Wang, Ma, Lu, & Wang, 2014). Analytical models of interconnection networks were considered in Nikitin & Cortadella, 2009; Kiasari, Lu & Jantsch, 2013; Rexford & Shin, 1996; and Kodgire & Shiurkar, 2015. Certain practical implementations were described in Chen et al., 2011; Liu & Carothers, 2011; Choudhary & Qureshi, 2012; Sadawarte, Gaikwad, & Patrikar, 2011; and Haag et al., 2015. Network latency in VCT networks is defined as the average time from the moment a message is generated by the source processor to the moment when the last flit of the message enters the consumption channel of the destination processor. The network latency consists of propagation delay, router delay, and contention (blockage) delay.

In this paper, we study network latency and saturation using the VCT routing policy. At each network node, we introduce (unlimited) storage buffers. The "unlimited" buffer model means in practice that the network throughput is limited by the link occupancy (utilization) rather than by the buffer capacity. So after the header is blocked, the "worm" collapses ("condenses") into the storage buffer. This method prevents deadlocks and improves network performance (increases bandwidth, decreases latency, delays saturation).

2. Communication Network Model

The following assumptions have been made for the network model implementation:

- The storage buffers are unlimited and use FIFO (first-in, first-out) policy.
- The same clock is used for all network nodes.
- When message is generated, it takes one time unit for its header (if not blocked) to appear at the router internal input port.
- It takes two time units (a time unit can include one or more clock cycles) to move a header flit from a router input port to its output port.
- It takes one time unit to move a flit (except the header) from a router input port to its output port.
- It takes one time unit to move a flit from the router output port to the input port of the next node (to go through the link).

2.1. Network Topology Model

In many systems where the VCT routing may be used, the physical distance between communication nodes is small and thus unimportant. In such systems, real network topology can be abstracted without loss of generality as easily constructed lower-dimension meshes and tori. This paper deals with two-dimensional torus networks. The symmetry of toroidal networks leads to a more balanced utilization of communication links than "open" mesh topologies and improves scalability.

Each node in such a network consists of a router and a local processor. Each router has four external input/output (I/O) channels, and one internal input/output channel to the local processor. All I/O channels are bidirectional so that two messages may travel simultaneously in the opposite directions between the nodes.

2.2. Routing

Each input/output port of the router contains three buffers: input buffer, output buffer and output storage buffer.

All input and output buffers can hold only one flit at a time. In our model, the storage buffer (which is an extension of the output buffer) is assumed to be "unlimited." This means that it can hold as many messages as needed at each moment of time. The local processor has the same one-flit sized input and output buffers. Suppose that the local processor generates message M_1 at time t_1, and another message M_2 at time t_2 ($t_2 > t_1$). It is possible that at time t_2 the router is still processing message M_1 and link from the local processor to the router is busy. To handle this situation, we introduce an unlimited storage buffer like the one at the external port, so the local processor can store the message M_2 in its output storage buffer until the link is free. As a result, external and internal ports have the same three-buffer architecture. This enabled us to use the same software implementation for external and internal ports.

It is believed that in real-life implementation the router processor needs more time (more time units) to route header flit (to decode destination, look up routing table, and decide on which output port to send) then to route remaining flits. We model this asymmetry by assigning two time units in the router for routing the header flit. Because all other flits of the message just follow the header, it takes one time unit in the router to send them to the correct output port.

Network contention occurs if two messages compete for the same channel. VCT networks generally outperform store-and-forward networks when the traffic patterns are sparse, but suffer substantial contention (leading to performance degradation) when the traffic is heavy. One way to address the issue of contention is to use adaptive routing, which allows a message to take an alternative path when primary path (defined in the routing table) is blocked by another message.

We use the deadlock-free adaptive unicast VCT routing algorithm as described below:

- Every router has a (static) two-dimensional routing table relating minimum length path from each of the router's four output ports to each network node.
- The routing table is used to perform dynamic routing based on the following set of rules:
  - If the current node (node to which the message header has arrived) is the destination, the header is routed to the internal port connecting to the local processor.
  - Else header of the message is sent from the input port to the output port, which has the shortest path to the destination node.
  - In general, more than one output port may have the minimum distance to destination, so the header is routed to the first available (free) port, where "first" refers to the port with the smallest number.
  - If all ports with minimum distance to the destination are busy, the router sends the header to the storage buffer of that one from these ports whose number is the largest one.
- If more than one simultaneously arrived headers should be routed to the same output port based on the rules described above, the header of the message with the smallest identification number will be processed first, and the header(s) of message(s) with larger identification number(s) will remain in the output storage buffer.
- Flits follow the header. If header motion is blocked, the header is routed to the storage buffer, all flits follow the header and accumulate (condense) in the storage buffer.

This routing algorithm assures that every link in the message path is occupied no longer than time equal to the message length.

2.3. Message Generation

We assume that at each time unit, every node in the network can generate a message with probability λ independently of all other nodes. Destination nodes for generated messages are

selected randomly among nodes having the specified distance l from the source node. Obviously, increasing λ increases the network load (the number of messages simultaneously traveling in the network), which, in turn, leads to the latency growth until network saturation is reached.

3. Theoretical Considerations

3.1. The Network States

We consider three different network states: startup, steady state, and saturation.

When network simulation starts up, initially there are no messages in the system. Then new messages start appearing in the network. Even in the absence of other messages, certain time τ_{min} is required for a message to reach its destination. During this time more messages can be generated, so initially after the startup the number of messages in the network increases. When the network reaches its steady state, the average number of messages generated during time Δt equals the average number of messages delivered during the same time Δt, so that the number of messages in the system (in transit from source to destination) becomes approximately constant over time. For a meaningful evaluation of the latency, the network must reach its steady state before data on the network behavior should be collected.

The network load increases with the number of messages present in the network and with the message length, since longer messages occupy links for longer time. As a result, for each message length, the network can accommodate only a limited range of message generation rates. If message generation rate is too high, the number of messages generated during time Δt exceeds the number of messages that can be delivered during this time, and the number of messages in the system is increasing with time with no bound. Thus, the latency tends to infinity, and network becomes dysfunctional. This is the state of saturation.

The rigorous theoretical analysis of the network performance with the VCT routing is a challenging problem that, as far as we know, has not yet been done even under the conditions of Jackson's theorem (1963). Here we restrict our task to deriving an expression for the saturation point (message generation rate at which network saturates) and approximate expressions for the latency.

3.2. Latency and Saturation Point

Let m be the length (the number of flits) of a message, n – the number of nodes, N – the number of messages in the network at a given moment of time, τ – the network latency, as defined in section 1, and ρ – the link utilization (the probability that a link is occupied during a unit time interval).

Let τ_{min} be the minimum time required for a message to reach the destination (the so-called base latency). Consider the case of very small λ (no blockages). A header needs one time unit to move from the output port of a router to the input port of the next one, two time units to

move from the input port to the output port in the same router and m time units to move all m flits through the consumption channel into the local processor at the destination node. It follows that $\tau_{min} = 3(l + 1) + m$. Obviously, in the case of contention, $\tau \geq \tau_{min}$.

Denote by L the total average number of links occupied by a message during all its average lifetime τ. Then, in the steady state, $N\frac{L}{\tau} = 4n\rho$, where $2n$ is the number of links (in both directions) in a 2-dim toroidal mesh. By Little's theorem, $N = \lambda n \tau$. Hence

$$\frac{\lambda L}{4} = \rho \tag{1}$$

Equation 1 is exact and is fulfilled for the whole range of values of λ, provided the network is in the steady state. In general, both L and ρ are functions of λ. Consider the case when λ is close to the critical value λ_{cr} that corresponds to saturation. Then $\rho \to 1$, which means almost no free links exist in the system. As a result, the delay at every router becomes very long, so that all flits of a message are condensed in the output buffer of the router and after that do not occupy any links. It follows that at any time interval at most one link is occupied by a message. Because of the "condensation", the total number of links occupied by a message during all the time the header spends in one router is $3 + (m - 3) = m$. Thus

$$\lim_{\rho \to 1} L = L_{\lim} = lm \tag{2}$$

By Equations 1 and 2, we obtain an expression for the critical value λ_{cr}

$$\lambda_{cr} = \frac{4}{lm} \tag{3}$$

In the spirit of Jackson's theorem (1963), let us assume that all events when a link is free or occupied are independent. (This assumption is analogous to the "mean field theory" in statistical physics). Then the probability $p(i)$ of delay i due to blockage is

$$p(i) = (1 - \rho)\rho^i \quad (i = 0, 1, 2, \ldots) \tag{4}$$

The average time delay $d(\rho)$ in a router due to contention is

$$d = d(\rho) = (1 - \rho)\sum_{i=0}^{\infty} i\rho(i) = \frac{\rho}{1-\rho} \tag{5}$$

An approximation for latency τ as a function of the generation rate λ can be obtained based on following analysis.

Let message length m satisfy inequality $m \geq 2l + 2$. Let us assume for the moment that d is the exact value of the blocking delay at every router k (k is the number of links from the source to the router). Then the general expression for the latency is

$$\tau = (l + 1)(d + 3) + m \tag{6}$$

The details of the propagation of the message along its path are different for three different ranges of values of *d*.

1. Small d: $d \leq \frac{m+1}{l+1} - 3$
2. Intermediate d: $\frac{m+1}{l+1} - 3 < d < m - 2$
3. Large d: $d \geq m - 2$

Consider these three cases separately:

1. $d \leq \frac{m+1}{l+1} - 3$. Then, at the end of the period of $d+3$ time units when the header remains in router k, the message occupies all k links. Because of condensation there are $d + 2$ flits in the router k and $d + 3$ flits in routers from 0 to $k - 1$. The number of flits remaining in the local processor of the source is $m - (k + 1)(d + 3) + 1$. It is readily seen that the total number L of links occupied by the message during its lifetime is
$$L = (d + 3)\sum_{k=1}^{l} k + l[(m - (l + 1)(d + 3) + 1] + (d + 3)\sum_{k=1}^{l} k - l = lm$$
2. $\frac{m+1}{l+1} - 3 < d < m - 2$. The dynamics of propagation is different in this case. Let k_d be the number of the router for which $k_d = \frac{m-d-2}{d+3}$. Then, the message always occupies k links when the header is in the router k if $k \leq k_d$. The message occupies $k_d + 1$ links during $d+2$ time units and k_d links during one time unit, when the header is in router k, $k_d + 1 \leq k \leq l$. It follow that the total number L of occupied links is
$$L = (d + 3)\sum_{k=1}^{k_d} k + (d + 2)(l - k_d) + (l - k_d)k_d + (d + 3)\sum_{k=1}^{k_d} k - k_d = lm$$
3. $d \geq m - 2$. In this case, all flits condense in one router before the header propagates to the next router. As discussed above, the total number of occupied links is the same, as in two other cases: $L = lm$.

Thus, in general, the total number of links occupied by a message during all its average lifetime τ does not depend on λ:

$$L = lm \tag{7}$$

In fact, the blockage delay is not a constant, but a random variable i distributed according to (4), with the expected value given by Equation 5: $d = \frac{\rho}{1-\rho}$. Hence, the average delivery time (the latency) τ, being a linear function of the delay, can be expressed as

$$\tau = (l + 1)(d + 3) + m = (l + 1)\left(\frac{\rho}{1-\rho} + 3\right) + m \tag{8}$$

Substituting $L = lm$ in Equation 1, we obtain:

$$\rho = \frac{\lambda l m}{4} = \frac{\lambda}{\lambda_{cr}} \tag{9}$$

It follows from (8) and (9) that

$$\tau = (l+1)\left(\frac{\lambda l m}{4-\lambda l m} + 3\right) + m \tag{10}$$

Expression 9 shows that transition to saturation is a second-order (continuous) phase transition with a critical exponent equal to 1, in agreement with the "mean field" theory. Note that for small λ ($\lambda \ll \frac{4}{lm}$), the latency is a linear function of λ and depend linearly also on the length of the message m, while the dependence on the distance l has a small quadratic term. For illustration purposes it is convenient to use expression (8) that yields the latency in terms of utilization ρ. The plots of τ as function of ρ with m and l as parameters are given in Figure 1.

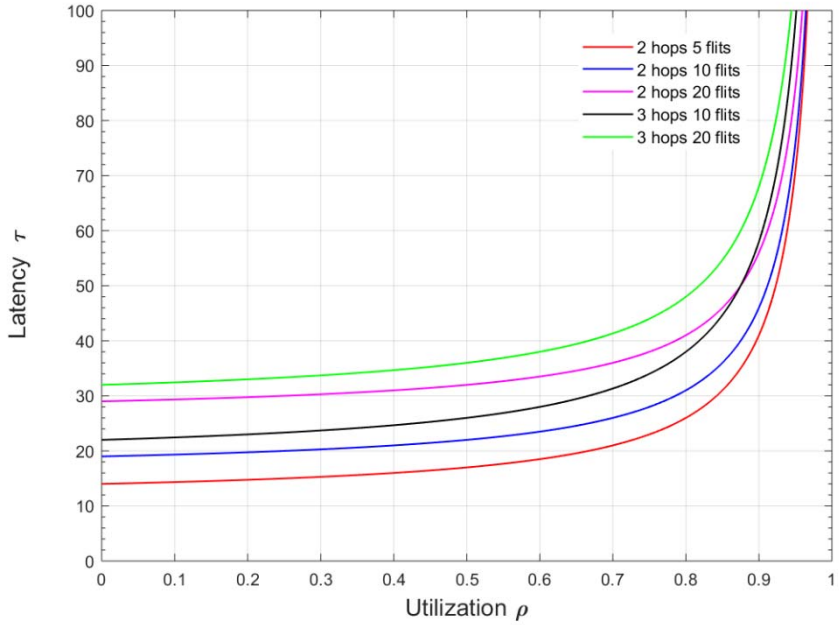

Figure 1. Theoretical approximation of latency τ as a function of utilization ρ. l = 2, 3 hops, m = 5, 10, 20 flits.

Consider now which buffer capacity is sufficient to justify our idealization of "unlimited buffers". According to Little's theorem (Kleinrock, 1975), the average number of messages in the system under the steady-state condition is $N = \lambda n \tau$. Assuming the worst case, when all messages are collapsed in storage buffers, the average number of flits per buffer is $\frac{1}{4} m \lambda \tau$. Therefore, the required buffer size depends critically on how close is the working range of λ to λ_{cr}:

$$B \approx \frac{1}{4} m \lambda \tau = \frac{\lambda}{\lambda_{cr}}\left[\left(1 - \frac{1}{l}\right)\left(\frac{\lambda/\lambda_{cr}}{1-\lambda/\lambda_{cr}}\right) + \frac{m}{l}\right] \tag{11}$$

If e.g., $\lambda \leq 0.9\lambda_{cr} = \frac{1.8}{lm}$, then the buffer capacity B should be of order of

$B \approx 10.8\left(1+\frac{1}{l}\right)+0.9\frac{m}{l}$. If $\lambda \leq 0.99\lambda_{cr}$, then $B \approx 50\left(1+\frac{1}{l}\right)+\frac{m}{l}$.

4. **Simulation**

4.1. Simulation Procedure

The relatively small network sizes have been chosen in order to reduce the simulation time. The distances from source to destination and the message lengths varied for different experiments, but were assigned prior to the simulation and kept constant during simulation run.

The network performance is characterized by latency (average delivery time) as a function of the network load, and by its saturation load, which describes maximum network capacity. The latency is obtained by averaging delivery times for all messages generated during time T. The time T is selected according to the value of λ so that the total number of messages generated in the system during time T per each destination node will be about the same for all values of λ (this number is about 120 in our experiments). Since the average number of messages generated per unit of time is proportional to λ the following empirical formula has been used to estimate T: $T \approx 10\times 4l/\lambda$.

To ensure that the network is in its steady state during data collection period, the delivery time was recorded for messages generated within time interval (t_{min}, $t_{min} + T$), where t_{min} was sufficiently large.

4.2. Simulation Results

In each simulation run, we analyzed the number of messages in the network as a function of time to determine network state and make sure that the latency is computed using data collected in the steady state.

Message latency was calculated as a function of generation rate λ using message lengths of 5, 10 and 20 flits for mesh sizes ranging from $n = 4\times4$ to $n = 12\times12$ (network linear size $s = 4$, 6, 8, 12). To ensure steady state, $t_{min} = 50000$ was used.

Latency for small message length $m = 5$ are shown in Figure 2. The results demonstrate the latency dependence on the path length l.

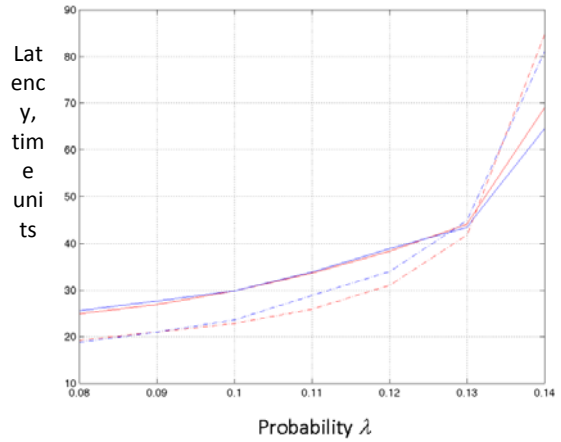

Figure 2. Latency for message length $m = 5$ flits. Mesh sizes 6×6 (blue lines) and 8×8 (red lines). Solid lines: $l = 3$, dashed lines: $l = 2$.

Simulations were performed using $l = 2$ and $l = 3$ with longer messages. Results for $l = 2$ and message length of $m = 10$ and $m = 20$ flits are shown in Figures 3 and 4, respectively. Since $s \geq 2l$ for $s = 6$, 8, 12 and $l = 2$, we expect similar network behavior, while the results for $d = 2$ are atypical ($s < 2l$) and demonstrate early saturation.

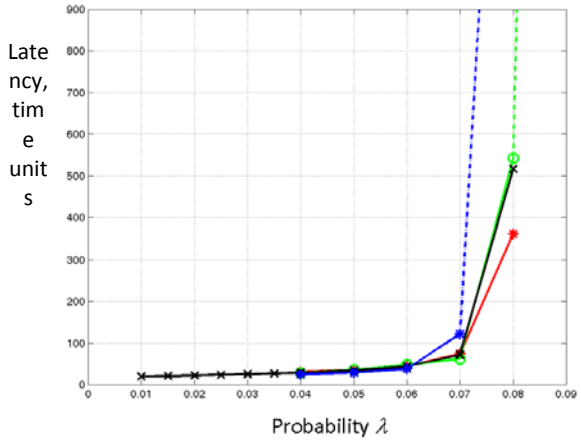

Figure 3: Latency for message length $m = 10$ flits. Path length $l = 2$ hops. Solid line: steady state; dashed line: steady state could not be reached. Mesh size 2×2 (blue line), 4×4 (red line), 6×6 (green line), and 8×8 (black line).

Simulation results for $l = 3$ are shown in Figures 5 and 6 for message lengths $m = 10$ and 20, respectively. Again, similar network behavior is observed, when $s \geq 2l$ condition is satisfied. These results indicate that network behavior (latency and saturation point) does not depend on the mesh size if the mesh linear dimension s is at least twice the message path length $s \geq 2l$. Results for latency dependence on the network load look somewhat paradoxical. It is seen that, if λ is small, latency is larger for longer paths. However, when $m = 5$, latency for the shorter path becomes larger for large values of λ, and saturation occurs earlier for the shorter path. Possibly, this behavior is related to the fact that the message length ($m = 5$) is

commensurate with the distance from the source to destination. Seemingly, this effect disappears if the length of the message is large enough (m = 20).

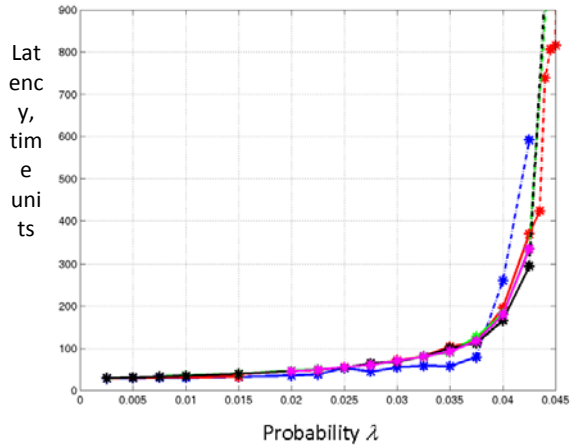

Figure 4. Latency for message length m = 20 flits. Path length l = 2 hops. Solid line: steady state; dashed line: steady state could not be reached. Mesh size 2×2 (blue line), 4×4 (red line), 6x6 (green line), 8×8 (black line), and 12×12 (magenta line).

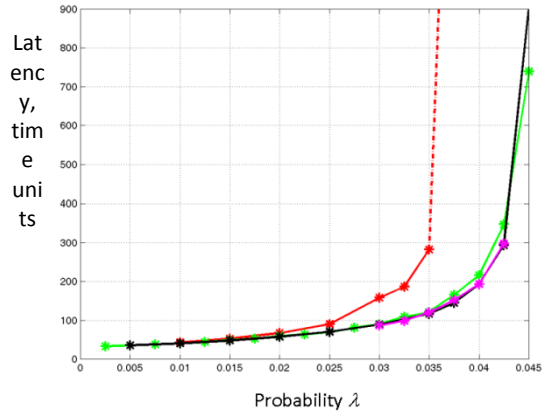

Figure 5. Latency for message length m = 10 flits. Path length l = 3 hops. Solid line: steady state; dashed line: steady state could not be reached. Mesh size 4×4 (red line), 6×6 (green line), 8×8 (black line).

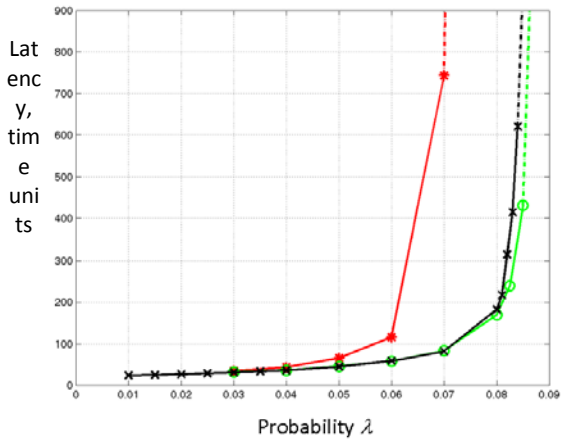

Figure 6: Latency for message length m = 20 flits. Path length l = 3 hops. Solid line: steady state; dashed line: steady state could not be reached. Mesh size 2×2 (blue line), 4×4 (red line), 6×6 (green line), 8×8 (black line), and 12×12 (magenta line).

4.3. Dependence of the Message Latency and Saturation on the Message Length

For the same message generation rate λ, longer messages result in higher network loads. Thus, we expect larger latencies and earlier saturation for longer messages.

Message latencies for the 2-hop path in 4×4, 6×6 and 8×8 meshes are shown in Figure 7 for message lengths m = 5 (blue), m = 10 (red), and m = 20 (green). Results are shown using star symbols (*) for 4×4 mesh, circles (○) for 6×6 mesh, and crosses (×) for 8×8 mesh. Similar results for 3 hops paths are shown in Figure 8.

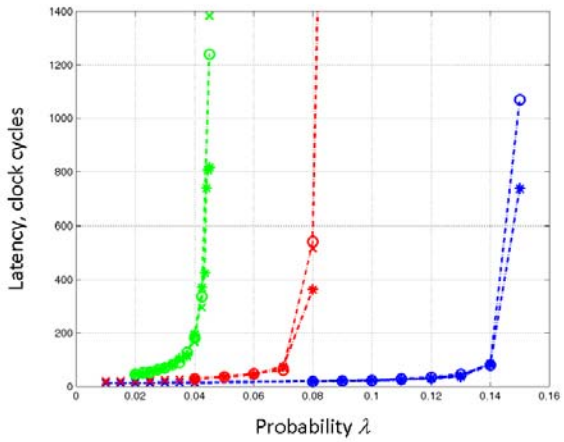

Figure 7. Latency as function of λ for l = 2 and various message lengths: m = 5 (blue lines), m = 10 (red lines), m = 20 (green lines). Mesh sizes: 4×4 (6 ,(*x6 (○), 8×8 (×).

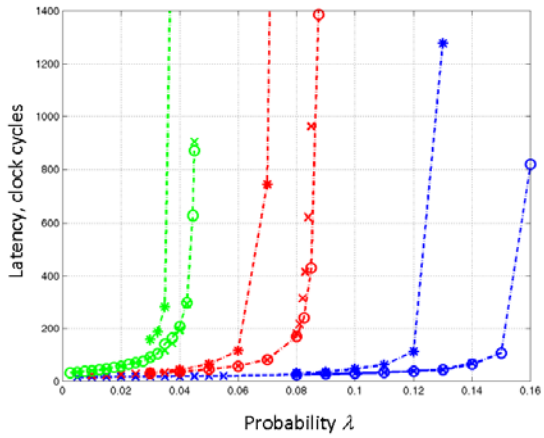

Figure 8. Latency as function of λ for $l = 3$ and various message lengths: $m = 5$ (blue lines), $m = 10$ (red lines), $m = 20$ (green lines). Mesh sizes: 4×4 (6 ,(*x6 (○), 8×8 (×).

Obviously, the network load increases with the message length. Let us hypothesize that the network load is proportional to the product of the message length m and the message generation rate λ. Then one can expect that the message generation rate at which network saturates (λ_{cr}) is inversely proportional to the message length m. The theoretical foundation for the hypothesis is given in Section 3.

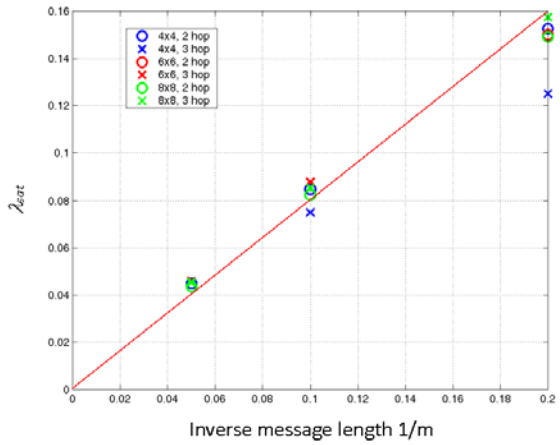

Figure 9. Message generation probability at which network saturates (λ_{cr}) as a function of the inverse message length $1/m$. Red line $\lambda_{sat} = 0.8/m$ shown for reference.

The results of our numerical experiments are in a good agreement with this hypothesis (see Figure 9). It is seen that for all cases, when $s \geq 2l$ the dependence of λ_{cr} of the message length m can be closely approximated as $\lambda_{cr} = 0.8/m$ (solid red line in Figure 9).

4.4. Dependence of the Message Latency and Saturation on the Message Length

In the steady state, the relationship between the average number of messages in the system and the latency is given by Little's theorem (Kleinrock, 1975) $N = \lambda n \tau$.

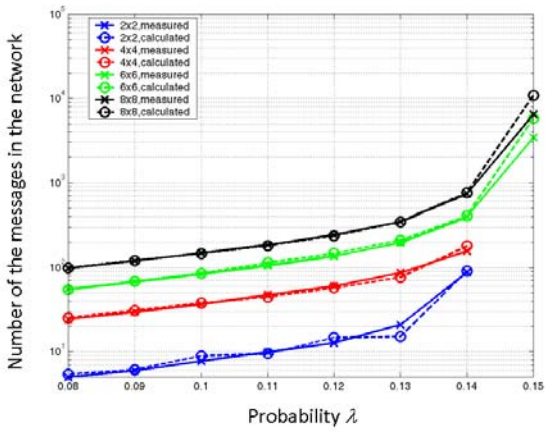

Figure 10. The number of the messages in the network as function of λ. Solid line: measured during simulation; dashed line: calculated by the use of Little's theorem. Message length m = 5 flits. Path length l = 2 hops.

Here N and τ are expected values of two random variables: number of messages in the network N_s, sampled over the total period of observation, and the sample delivery time τ_s. Therefore, the values of N_s and τ_s fluctuate with time and the relationship between N_s and τ_s satisfies Little's theorem only approximately.

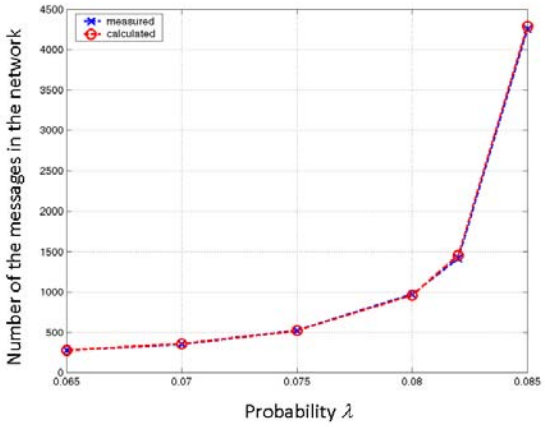

Figure 11. The number of messages in the network as function of λ. Mesh size 8×8. Message length m = 10 flits. Path length l = 3 hops. Solid line: measured, dashed line: calculated.

We have measured values of N_s by averaging the number of messages in the network from t_{min} up to the end of simulation, as well as calculated the number of messages in the network using the observed values of τ_s. As shown in Figures 10 and 11, the directly measured values of N_s and those calculated by the use of Little's theorem are in a very good agreement, which supports the validity of the simulation experiments. However, note that when network state is closed to saturation, the calculated number of messages usually exceeds the measured value.

5 Conclusion

A model of a 2-dimensional toroidal interconnection network with virtual cut-through routing has been studied. Analytical expression for the saturation point and approximate expressions for the network latency for the ranges of small network loads and loads close to the critical value have been obtained.

- The critical value of the probability of message generation $\lambda = \lambda_{cr}$ is inversely proportional to the distance between the source and the destination l and the length of messages m: $\lambda_{cr} = \frac{2}{lm}$.
- The latency τ at the saturation point experiences a second-order (continues) phase transition with the critical exponent equal to 1.
- For small values of λ, the latency grows as a linear function of λ.

Simulation experiments have been performed in order to find out and analyze certain empirical relationships that can be used as a starting point for a deeper theoretical analysis and further research. In particular, the following results have been obtained.

- Network behavior (latency and saturation point) does not depend on the mesh size if the mesh is "large enough" compared to the path length. As an appropriate criterion, the mesh linear dimension should be at least twice as large as the message path length: $s \geq 2l$.
- For the same message generation rate, latency increases and saturation occurs earlier for longer messages. It appears that the saturation point λ_{cr} (message generation rate at which network saturates) is inversely proportional to the message length. If the condition $s \geq 2l$ is satisfied, numerical results are in a good agreement with a simple empirical relation $\lambda_{cr} = 0.8/m$ independently of the mesh size. (It seems to be consistent also with the theoretical Equation 3).
- If the network is in the steady state, the independently measured number of messages N_s and the average delivery time τ_s are in a good agreement with Little's theorem for their expected values $N = \lambda n \tau$.

References

Chen, D., Eisley, N. A., Heidelberger, P., Senger, R. M., Sugawara, Y., Kumar, S., Salapura, V., …, Parker, J. J. (2011). The IBM Blue Gene/Q interconnection network and message unit. *Proceedings of the 2011 International Conference for High Performance Computing, Networking, Storage and Analysis*. New York: ACM.

Choudhary, S., & Qureshi, S. (2012). Performance evaluation of mesh-based NoCs: Implementation of a new architecture and routing algorithm. *International Journal of Automation and Computing*, *9*(4), 403-413.

Dabrowski, C. (2015, November). Catastrophic event phenomena in communication networks. *Computer Science Review*, *18*, 10-45. doi: https://doi.org/10.1016/j.cosrev.2015.10.001

Daf, M. P., & Saynkar, B. B. (2014). Performance and evaluation of loopback virtual channel router with heterogeneous router for on-chip network. *Proceedings of the 2014 Fourth International Conference on Communication Systems and Network Technologies*. Piscatawny, NJ: IEEE.

Domkondwar, P., & Chaudhari, D. (2012). Implementation of five port router architecture using VHDL. *International Journal of Advanced Research in Computer Science and Electronics Engineering*, *1*(3), 17-20.

Duato, J., Robles, A., Silla, F., & Beivide, R. (2001). A comparison of router architectures for virtual cut-through and wormhole switching in a NOW environment. *Journal of Parallel and Distributed Computing*, *61*(2), 224-253. doi: http://dx.doi.org/10.1006/jpdc.2000.1679

Hag, A. A. Y., Hafizur Rahman, M.M., Nor, R. M., Sembok, T. M. T., Miura, Y., & Inoguchi, Y. (2015). Uniform traffic patterns using virtual cut-through flow control on VMMN. *Procedia Computer Science*, *59*, 400-409. doi: http://dx.doi.org/10.1016/j.procs.2015.07.553

Jackson, P. (1963). Job shop like queueing systems. *Management Science*, *10*(1), 131-142.

Karpovsky, M. G., Levitin, L. B., & Mustafa, M. (2014). Optimal turn prohibition for deadlock prevention in networks with regular topologies. *IEEE Transactions on Control of Network Systems*, *1*(1), 74-85.

Kaushal, A., & Singh, S. (2014). Network on chip architecture and routing techniques: A survey. *International Journal of Research in Engineering and Science*, *2*(6), 65-79.

Kermani, P., & Kleinrock, L. (1979). Virtual cut-through: A new computer communication switching technique. *Computer Networks*, 3(2), 267-286.

Kiasari, A. E., Lu, Z., & Jantsch, A. (2013). An analytical latency model for networks-on chip. *IEEE Transactions on Very Large Scale Integration Systems (VLSI)*, *21*(1), 113-123.

Kleinrock, L. (1975). *Queueing systems volume I: Theory*. New York: Wiley.

Kodgire, S., & Shiurkar, U. (2015). An analytical router model for networks-on-chip. *IOSR Journal of VLSI and Signal Processing*, *5*(4), 16-21.

Lekariya, D., & Gaikwad, M.A. (2013). Performance analysis of minimum hop source routing algorithm for two dimensional torus topology NOC architecture under CBR traffic. *IOSR Journal of Electronics and Communication Engineering*, *7*(2), 5-12.

Levitin, L. B., & Rykalova, Y. (2015). Analysis and simulation of computer networks with unlimited buffers. In A.-S. Pathan, M. Monowar, & S. Khan (Eds.). *Simulation Technologies in networking and communications: Selecting the best tool for the test* (pp. 3-300). St. Paul, MN: CRC Press.

Levitin, L. B., & Rykalova, Y. (2016). Latency and phase transitions in interconnection networks with unlimited buffers. *International Journal of Modern Engineering, 17*(1),70-77.

Levitin, L. B., Karpovsky, M. G., & Mustafa, M. (2006). A new algorithm for finding minimal cycle-breaking sets of turns in a graph. *Journal of Graph Algorithms and Applications*, *10*(2), 387-420.

Levitin, L. B., Karpovsky, M. G., & Mustafa, M. (2009). Minimal sets of turns for breaking cycles in graphs modeling networks. *IEEE Transactions Parallel and Distributed Systems*, *21*(9), 1342-1353.

Levitin, L. B., Karpovsky, M. G., & Mustafa, M. (2013). Deadlock prevention in networks of workstations with wormhole routing: irregular topology. In A. Loo (Ed.). *Distributed innovations for business, engineering and science* (pp. 47-73). Hershey PA: IGI.

Li, W., Bashan, A., Buldyrev, S. V., Stanley, H. E., & Havlin, S. (2012). Cascading failures in interdependent lattice networks: The critical role of the length of dependency links. *Physical Review Letters*, *108*(22), 228702.

Liu, N., & Carothers, C. D. (2011). Modeling billion-node torus networks using massively parallel discrete-event simulation. *IEEE Workshop on Principles of Advanced and Distributed Simulation*. Piscatawny, NJ: IEEE. doi: 10.1109/PADS.2011.5936761

McCarthy, C. M., Isaacs, K. E., Bhatele, A., Bremer, P.-T., & Hamann, B. (2014). Visualizing the five-dimensional torus network of the IBM blue gene/Q. *Proceedings of the First Workshop on Visual Performance Analysis*. Piscataway, NJ: IEEE. doi: http://dx.doi.org/10.1109/VPA.2014.10J

Minkenberg, C. (2013, May 21). *Interconnection network architectures for high-performance computing. Advanced computer networks*. [PowerPoint slides]. Retrieved from https://manualzz.com/doc/13508828/interconnection-network-architectures-for-high-performance-computing

Nikitin, N., & Cortadella, J. (2009). A performance analytical model for network-on-chip with constant service time routers. *Proceedings of the 2009 International Conference on Computer-Aided Design*. New York: ACM.

Opoku Agyeman, M., Zong, W., Yakolev, A., Tong, K.-F., & Mak, T. (2017). Extending the performance of hybrid NoCs beyond the limitations of network heterogeneity. *Journal of Low Power Electronics and Applications*, *7*(2), 8.

Rexford, J., & Shin, K. G. (1996). *Analytical modeling of routing algorithms in virtual cut-through networks*. Ann Arbor, MI: University of Michigan Press.

Rykalova, Y., Levitin, L. B., & Brower, R. (2010). Critical phenomena in discrete-time interconnection networks. *Physica A: Statistical Mechanics and Its Applications*, *389*(22), 5259-5278.

Rykalova, Y., & Levitin, L. (2017). Phase transitions in interconnection networks with finite buffers. *International Journal of Modern Engineering*, *17*(2), 33-40.

Sadawarte, Y. A., Gaikwad, M. A., & Patrikar, R. M. (2011). Implementation of virtual cut-through algorithm for network on chip architecture. *IJCA Proceedings on International Symposium on Devices MEMS, Intelligent Systems & Communication*, *1*, 5-8.

Wang, P., Ma, S., Lu, H., & Wang, Z. (2014). A comprehensive comparison between virtual cut-through and wormhole routers for cache coherent network on-chips. *IEICE Electronics Express*, *11*(14), 20140496-20140496.

Biographies

LEV LEVITIN is currently distinguished professor of Engineering Science with the Department of Electrical and Computer Engineering at Boston University, Boston, MA. He has published over 200 papers, presentations, and patents. His research areas include information theory; quantum communication systems; physics of computation; quantum computing; quantum theory of measurements, mathematical linguistics; theory of complex systems; coding theory; theory of computer hardware testing, reliable computer networks,

and bioinformatics. He is a life fellow of IEEE, a member of the International Academy of Informatics and other professional societies. Prof. Levitin may be reached at levitin@bu.edu.

YELENA RYKALOVA is currently teaching at UMass Lowell, MA, and visiting researcher with the Department of Electrical and Computer Engineering at Boston University, Boston, MA. Her research interests are in computer networks, in particular, in application of concepts and models of statistical physics to the analysis of network performance. She is a member of IEEE and the Society for Modeling and Simulation International. Since 2008, she has been active in organization and preparation for the *Spring Simulation Multiconference* as technical committee member, reviewer, and publicity and session chair. Dr. Rykalova may be reached at yelena_rykalova@uml.edu.