A SURVEY TO REAL-TIME MESSAGE-ROUTING NETWORK SYSTEM WITH KLA MODELLING

Paul T. Chiou, Yu Sun and G S. Young

Department of Computer Science, California State Polytechnic University, Pomona, California USA

ABSTRACT

Messages routing over a network is one of the most fundamental concept in communication which requires simultaneous transmission of messages from a source to a destination. In terms of Real-Time Routing, it refers to the addition of a timing constraint in which messages should be received within a specified time delay. This study involves Scheduling, Algorithm Design and Graph Theory which are essential parts of the Computer Science (CS) discipline. Our goal is to investigate an innovative and efficient way to present these concepts in the context of CS Education. In this paper, we will explore the fundamental modelling of routing real-time messages on networks. We study whether it is possible to have an optimal on-line algorithm for the Arbitrary Directed Graph network topology. In addition, we will examine the message routing's algorithmic complexity by breaking down the complex mathematical proofs into concrete, visual examples. Next, we explore the Unidirectional Ring topology in finding the transmission's "makespan".Lastly, we propose the same network modelling through the technique of Kinesthetic Learning Activity (KLA). We will analyse the data collected and present the results in a case study to evaluate the effectiveness of the KLA approach compared to the traditional teaching method.

KEYWORDS

Real Time Routing, Routing Messages, Network Routing, Graph Theory, Scheduling, Arbitrary Directed Graph, Unidirectional Ring, Kinesthetic Learning Activity, KLA, CS Education

1. INTRODUCTION

In the age of multimedia today, almost everything is transferred through the Internet. As the quality of multimedia rapidly increases over the past decade, the demand for a faster network bandwidth is also growing at a significant rate. There are many applications relying on real-time routing services. For instance, in a live messaging system, each participant interacts with the others by sending and receiving messages in the form of texts, pictures, audio sequences, or video frames [1]. Each message, originating from a participant, must be delivered to all the others in a real-time manner. All these real-time systems are constantly in need to optimize their Quality of Service provided to the users [2].The transfer speed is bounded by the network infrastructure itself in the hardware level. However, it is essential to theoretically understand how these routing take place so we can attempt to optimize for a more efficient algorithm. This is indeed an area of research for CS students that crosses multiple domains such as Scheduling, Algorithm Design, Complexity Analysis, NP Theory, Graph Theory, etc. According to the ACM and IEEE, CS Education is a pathway to innovation and creativity that requires an understanding of the fundamentals problem-solving methodology of computational thinking [3]. In our research, we will introduce the model of a real-time system and present its method for scheduling messages.We shall then propose and incorporate our own modelling technique using the innovative non-traditional KLA to attempt a more effective learning approach for CS Education.

2. INTRODUCTION TO REAL-TIME MESSAGE ROUTING

As mentioned, in a distributed real-time system, the problem of determining whether a set of messages can be sent on time becomes an important issue. In the last two decades, real-time message routing has been studied extensively on various types of networks [4]. An on-line routing algorithm is one that routes messages without any knowledge of future arrivals of messages. Based on the different level of constraints, the complexities can vary from easy (being solvable in polynomial time) while others are NP-hard [5]. Next, we shall first introduce the fundamental notion of various restrictions of the four parameters- origin node, destination node, release time, and deadline.

2.1. The Formal Model and Notation

Wewill follow the scheduling standard and represent a network by a directed graph $G = (V, E)$ shown in *Figure 1*, where each vertex in the set V represents a node of the network and each directed edge in set E represents a communication link. If $(u, v) \in E$, then there is a transmitter in node u and a receiver in node v dedicated to the communication link (u, v) in *Figure 2*. We assume that a node can simultaneously send and receive several messages if they are transmitted on different communication links [6].
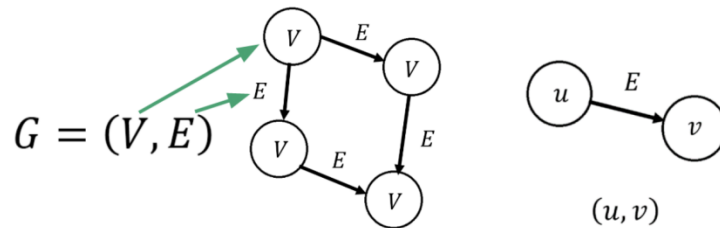

Figure 1. Network as a Directed Graph (Left),Communication Link (Right)

In our rounding problem, a set of n messages $M = (M_1, \ldots, M_n)$ as shown in *Figure 3* needs to be routed through the network. Each message M_i consists of its own characteristic properties represented by a quintuple $(s_i, e_i, l_i, r_i, d_i)$. In this set, the s_i denotes the *origin node*(i.e., M_i originates from node s_i), e_i denotes the *destination node* (i.e., M_i is to be sent to node e_i), l_i denotes the *length* (i.e., M_i consists of l_i packets of information), r_i denotes the *release time* (i.e., M_i originates from s_i at time r_i), and d_i denotes the *deadline* (i.e., M_i must reach e_i by time d_i). These features are the constraints in which we will follow to vary the message routing complexity. The examples of these message routing problems consist of a network G and a set of messages M. In our model, we further assume that it takes one unit of time to send a packet through a communication link. This implies that it takes length l_i time units for message M_i to traverse on any communication link. Also, there exists a central controller which has complete information on the topology of the network and the characteristics of each messages in advance. Our goal is to determine if the messages in M can be routed through the network G such that each message M_i is sent from node s_i to node e_i satisfying the given time interval of $[r_i, d_i]$ [5], [6].

2.2. Message-Routing Network System

A message-routing network system is a given system that consists of the network itself and the messages to be transmitted across it. In notation, we let $MRNS = (G, M)$ be the "Message-

Routing Network System", where $G = (V, E)$ is a network and $M = (M_1, M_2, \ldots, M_n)$ is a set of n messages to be routed through G. [5], [6].

A transmission S for $MRNS$ is said to be feasible if the deadlines of all the messages are met; i.e., for all $q = (u, v, M_i, t_1, t_2), t_2 \leq d_i$. Here, t_2 is the finishing time of the communication link. A non-preemptive transmission, denoted by S_{NP}, is a transmission such that for each configuration$(u, v, M_i, t_1, t_2) \in S_{NP}, t_2 - t_1 = l_i$. A preemptive transmission, denoted by S_P, is a transmission defined exactly as before. A $MRNS$ is feasible with respect to non-preemptive (preemptive) transmission if there is a feasible non-preemptive (preemptive) transmission for $MRNS$ [6].

3. REAL TIME ROUTING COMPLEXITY ANALYSIS

3.1. Arbitrary Directed Graph

In this section, we present a Message-Routing Network System $MRNS$with an arbitrary directed graph as shown in the *Figure 2*. We let $MRNS = (G, M)$, where G is a network given and $M = \{M_1, M_2, M_3, M_4\}$. The four messages to be transferred are defined with the following constraints: $M_1 = (1, 4, 3, 0, 6), M_2 = (1, 4, 2, 2, 6), M_3 = (3, 5, 2, 0, 6)$, and $M_4 = (4, 5, 4, 0, 4)$. The table in *Figure 9* shows an organized view. Recall the quintuple defined $(s_i, e_i, l_i, r_i, d_i)$ which represents the origin node, destination node, length, release time, and deadline [5, 6].
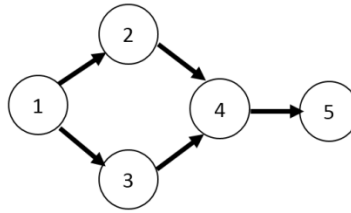

Figure 2. Arbitrary Directed Graph Network

M_i	s_i origin node	e_i destination node	l_i length	r_i release time	d_i deadline
M_1	1	4	3	0	6
M_2	1	4	2	2	6
M_3	3	5	2	0	6
M_4	4	5	4	0	4

Table 1. Message Constraints of M for the Arbitrary Directed Graph$MRNS$.

3.1.1. Non-Preemptive Transmission

First, we will examine a non-preemptive instance of the $MRNS$. In non-preemptive transmission, once a message is transmitted on a communication link (u, v), it must continue until the entire message is received by node v. We use the subscript NP to denote the transmission S is non-preemptive. With the message constraints in *Table 1*, we can construct the following transmission

$S_{NP} = \{(1, 2, M_1, 0, 3), (2, 4, M_1, 3, 6), (1, 3, M_2, 2, 4), (3, 4, M_2, 4, 6), (3, 4, M_3, 0, 2),$
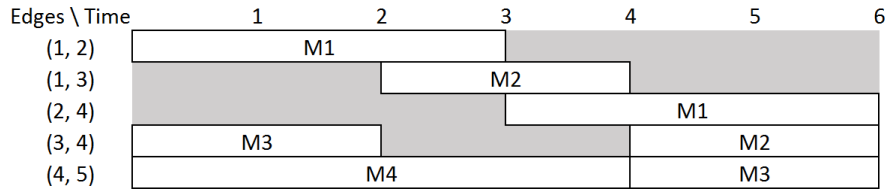$(4, 5, M_3, 4, 6), (4, 5, M_4, 0, 4)\}$, as shown in *Figure 3* [6].

Figure 3. Feasible Non-Preemptive Transmission for $MRNS$.

M_1, M_2 are departing from their origin node 1 to reach destination node 4. For this transmission, M_1 will take route $1 \rightarrow 2 \rightarrow 4$ and M_2 will take route $1 \rightarrow 3 \rightarrow 4$ since this will be more efficient having both messages transferred simultaneously during time 2 to 3, 3 to 4, and 4 to 6. Note that at time 0, messages M_1, M_3, M_4 are released from their origin nodes $1, 3, 4$ respectively. Each of the corresponding message lengths are $3, 2, 4$. Therefore, messages M_1, M_3, M_4 arrives at node $2, 4, 5$ at times $3, 2, 4$. Since M_4's destination node is 5, its transfer is then complete at time 4. While M_1, M_3, M_4 are transferring at time 2, M_2 is ready to be released and it takes 2 units (its length) time for it arrive at node 3. At time 3, when M_1 finishes arriving at node 2, it can begin its transfer to its destination node 4. At time 4, when M_2 finishes arriving at node 3, it can begin its transfer to its destination node 4. Note that the communication link $(4, 5)$ is being occupied by M_4 so M_3 must wait until it finishes (considering this is a non-preemptive transmission). Finally, M_3 begin its transfer to its destination node 5 along with M_1, M_2 [6].

Clearly, S_{NP} is a feasible non-preemptive transmission because all messages met their deadlines. Thus, with this example, we can conclude this $MRNS$ is feasible with respect to non-preemptive transmission.

3.1.2. Preemptive Transmission

Unlike non-preemptive where all messages must finish its transfer once it starts, preemptive transmission can be interrupted and resume later. This provides more flexibility and can be more efficient although it adds complexity to the problem. In fact, if a $MRNS$ is feasible with non-preemptive transmission, it is also feasible with preemptive transmission [6].

In the following example, we will slightly modify the previous instance to demonstrate the preemptive transmission case. Consider a new system $MRNS'$ obtained from $MRNS$ by modifying M_4 to be $(4, 5, 2, 3, 5)$. Now M_4 has a shorter message length of 2 and later deadline of 5. However, its released time is postponed to 3. *Table 2* shows the change in grey.

M_i	s_i origin node	e_i destination node	l_i length	r_i release time	d_i deadline
M_1	1	4	3	0	6
M_2	1	4	2	2	6
M_3	3	5	2	0	6
M_4	4	5	2	3	5

Table 2. Modified Message Constraints of M for $MRNS'$

This time in $MRNS'$, messages M_1, M_2 remains unchanged as M_4 does not affect them. However, M_4 is constrained and cannot be released until time 3. When M_3 finishes transferring to node 4 at time 2, it can begin its transfer to its destination node 5 without being idling between time 2 to 3. Now at time 3, M_4 is released and since this is preemptive transmission, M_3 can be interrupted to continue later. Hence, M_4 begins and finishes arriving at its destination node 5

after 2 time units(its length). Finally, the interrupted M_3 can now resume its transfer, completing at destination node 5attime6.Thetransmission

$S'_P\{(1,2,M_1,0,3),(2,4,M_1,3,6),(1,3,M_2,2,4),(3,4,M_2,4,6),(3,4,M_3,0,2),(4,5,M_3,2,3),$
$(4,5,M_3,5,6),(4,5,M_4,3,5)\}$, is a feasible preemptive transmission. However, the modification made it not feasible with respect to non-preemptive transmission. By examining *Figure 4*, we see the reason is due to M_3. At time 3 when M_4 is ready to be released, without preemption, M_3 must keep transferring until it finishes at time 4 occupying communication link $(4,5)$ and M_4 must wait. When M_4 is finally able to begin transfer at time 4, its message length of 2 will result a finishing time of 6, therefore missing its deadline at time 5. Thus $MRNS'$ is feasible with respect to preemptive transmission, but not non-preemptive transmission [5], [6].
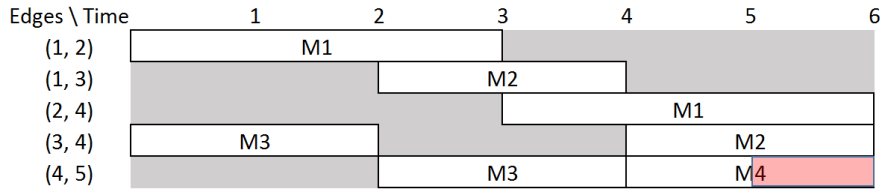

Figure 4. Non-feasible Preemptive Transmission for $MRNS'$

3.1.3. Theorem and Proof

We now show that if the network is an arbitrary directed graph, the problem of determining whether there is a feasible transmission (both preemptive or non-preemptive) for a message-routing network system is NP-complete. The hardness is not caused by the message constraints but the nature of the network topology because it is NP-complete even when all four parameters are fixed [6]. Consider the following theorem:

3.1.3.1 Theorem 37.1

Given $MRNS = (G, M)$, where G is an arbitrary directed graph and M is a set of messages with identical origin nodes, destination nodes, release times, and deadlines, the problem of determining whether MRNS is feasible with respect to preemptive transmission is NP-complete [6].

The theorem can be proved by the reduction from the 3-Partition problem, which is known to be strongly NP-complete. Before showing the proof, we will explore the 3-Partition problem.The 3-partition problem is a problem where one should partition $3z$ numbers (allowing duplicates) into z groups of 3, such that each group has the same sum. If all $3z$ numbers sum to N, this means that every group should have a sum of N/z. The 3-partition problem is a well-known NP-complete problem. Next, we use formal notation to precisely define the problem [6, 7].

Given a list, $A = (a_1, a_2, \dots, a_{3z})$ of $3z$ positive integers such that$\sum_{i=1}^{3z} a_i = zB$and each a_i satisfying$\frac{B}{4} < a_i < \frac{B}{2}$ for each $1 \leq i \leq 3z$, can $I = \{1, 2, \dots, 3z\}$ be partitioned into z groups of 3, $I_1, I_2, \dots, I_z$ such that $\sum_{i \in I_j} a_i = B$ for each $1 \leq j \leq z$ (each group sums to exactly B).

The 3-partition problem is similar to the partition problem, which in turn is related to the subset sum problem. In the partition problem, the goal is to partition S into two subsets with equal sum. In 3-partition the goal is to partition S into z subsets, not just two subsets, with equal sum [7].

To show that the decision problem is in NP, notice that preemption is necessary only when a new message arrives at a node. Thus, the number of preemptions is bounded by a polynomial function

of the size of the input, and hence we can guess a preemptive transmission in polynomial time and verify that the transmission is feasible [6]. To complete the proof, the 3-Partition problem reduction can be defined as follows:

Let $A = (a_1, a_2, ..., a_{3z})$ be an instance of the 3-Partition problem. We construct an instance of the message routing problem as follows: $MRNS = (G, M)$, where

$$G = \left(\left\{ \begin{array}{c} \{0, 1, 2, ..., z + 1\}, \\ (0,1), (0,2), ..., (0,z), \\ (1, z+1), (2, z+1), ..., (z, z+1) \end{array} \right\} \right)$$

and $M = (M_1, M_2, ..., M_{4z})$, and $M_i = (0, z+1, a_i, 0, 5B)$ for each $1 \leq i \leq 3z$ and $M_i = (0, z+1, 2B, 0, 5B)$ for each $3z + 1 \leq i \leq 4z$. We call the first $3z$ messages the partition messages and the remaining z messages the enforcer messages [5]. This is visually represented in *Figure 5*

Figure 5. Visual Representation of G and M for 3-Partition $MRNS$

Clearly, based on the definition, all messages in M have identical origin nodes, destination nodes, release times, and deadlines. Thus, construction can be done in polynomial time. Since the hard 3-partition is given by the oracle, the hard-part is done. The polynomial construction simply involves calculating the extended message length and mapping the nodes. Adding the enforcers is also be done in polynomial time [6].

For the reduced instance, there are z distinct paths from vertex 0 to vertex $z + 1$ in G. In any feasible transmission for $MRNS$, each enforcer message must be routed along a distinct path from vertex 0 to vertex $z + 1$. The remaining partition messages must be distributed among these z paths [6]. This is intuitively correct because the enforcer messages essentially "jams up" each of the route, reducing the complexity and possibilities of combinatorically assigning the partition messages.

The above $MRNS$ is feasible with respect to preemptive transmission if and only if A has a 3-Partition. We now show the case with a substantial instance of the reduction process. The following is a "yes"' instance of the 3-partition of $3z$ messages with $z = 3$. Notice that our $B = 100$, which is the sum of each partition with the elements satisfying size $\frac{100}{4} < a_i < \frac{100}{2}$. Otherwise, it would be impossible to have 3 parts each [7].

$$A = \{a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9\} = \{26, 29, 33, 33, 33, 34, 35, 36, 41\}$$
$$I_1 = \{a_1, a_3, a_9\} = \{26, 33, 41\}$$
$$I_2 = \{a_2, a_8, a_7\} = \{29, 36, 35\}$$
$$I_3 = \{a_4, a_5, a_6\} = \{33, 33, 34\}$$

These $3z$ elements maps to the corresponding partition messages. We then add z more messages M_{10}, M_{11}, M_{12} as the enforcer messages. The set M now consists of $4z$ messages:

$$M = \{M_1, M_2, M_3, M_4, M_5, M_6, M_7, M_8, M_9, M_{10}, M_{11}, M_{12}\}$$

M_i	s_i origin node	e_i destination node	l_i length	r_i release time	d_i deadline
M_1	0	4	26	0	500
M_2	0	4	29	0	500
M_3	0	4	33	0	500
M_4	0	4	33	0	500
M_5	0	4	33	0	500
M_6	0	4	34	0	500
M_7	0	4	35	0	500
M_8	0	4	36	0	500
M_9	0	4	41	0	500
M_{10}	0	4	200	0	500
M_{11}	0	4	200	0	500
M_{12}	0	4	200	0	500

Figure 6. Message Constraints of M for 3-Partition $MRNS$

Suppose $I_1, I_2 \ldots, I_z$ is a solution to the 3-Partition problem. We can construct a feasible non-preemptive (and hence preemptive) transmission as follows [6].

- For each $1 \leq j \leq z$, the messages M_{3z+j} and M_i, where $i \in I_j$, are transmitted from node 0 to node $z + 1$ through the edges $(0, j)$ and $(j, z + 1)$. Shown in *Figure 7*.
- M_{3z+j} is transmitted through the edges $(0, j)$ and $(j, z + 1)$ in the time intervals $[0, 2B]$ and $[2B, 4B]$ respectively. Shown in *Figure 8 (left)*.
- The messages M_i, where $i \in I_j$, are nonpreemptively transmitted through the edges $(0, j)$ and $(j, z + 1)$ in the time intervals $[2B, 3B]$ and $[4B, 5B]$, respectively. Shown in *Figure 8 (right)*.

Using the above algorithm, we created a feasible non-preemptive transmission using the properties of our 3-partition instance as follows

$$1 \leq j \leq 3$$
$$I_1 = \{a_1, a_3, a_9\} = \{M_1, M_3, M_9\} \quad M_{3(3)+1} = M_{10}$$
$$I_2 = \{a_2, a_8, a_7\} = \{M_2, M_8, M_7\} \quad M_{3(3)+2} = M_{11}$$
$$I_3 = \{a_4, a_5, a_6\} = \{M_4, M_5, M_6\} \quad M_{3(3)+3} = M_{12}$$

Figure 7. Message Categorization of the Partitions.

- For $j = 1$, M_{10}, M_1, M_3, M_9 are transmitted from node 0 to node 4 through the edges $(0, 1)$ and $(1, 4)$ with M_{10} in the time intervals $[0, 200]$ and $[200, 400]$ and M_1, M_3, M_9 non-preemptively transmitted in the time intervals $[200, 300]$ and $[400, 500]$
- For $j = 2$, M_{11}, M_2, M_8, M_7 are transmitted from node 0 to node 4 through the edges $(0, 2)$ and $(2, 4)$ with M_{11} in the time intervals $[0, 200]$ and $[200, 400]$ and M_2, M_8, M_7 non-preemptively transmitted in the time intervals $[200, 300]$ and $[400, 500]$
- For $j = 3$, M_{12}, M_4, M_5, M_6 are transmitted from node 0 to node 4 through the edges $(0, 3)$ and $(3, 4)$ with M_{12} in the time intervals $[0, 200]$ and $[200, 400]$ and M_4, M_5, M_6 non-preemptively transmitted in the time intervals $[200, 300]$ and $[400, 500]$
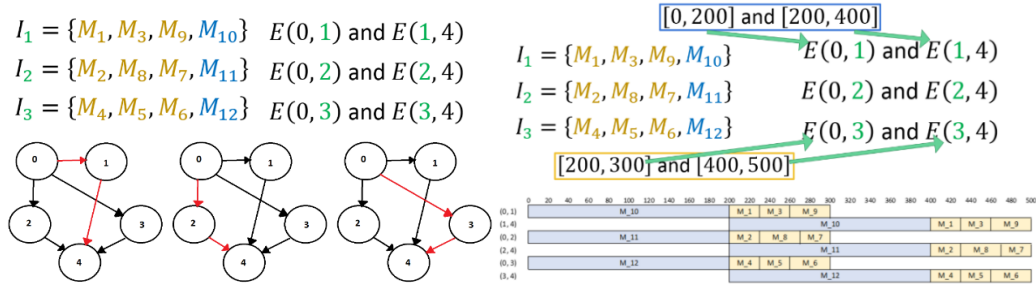
Figure 8. Messages' Distinct Paths (Left), and Transmission Times (Right)

Path distinctness in the proposed $MRNS$ can be show by supposing S is a feasible preemptive transmission for $MRNS$. Observe on the left of *Figure 8* that there are exactly z distinct paths from node 0 to node $z + 1$; namely, $P_1 = (0, 1, z + 1)$, $P_2 = (0, 2, z + 1)$, ..., and $P_z = (0, 3, z + 1)$ [6]. In our case, there are exactly 3 distinct paths from node 0 to node 4 which are $P_1 = (0, 1, 4)$, $P_2 = (0, 2, 4)$, and $P_3 = (0, 3, 4)$.

With these distinct paths, each message must be transmitted through one of these z paths and if two enforcer messages are transmitted through the same path in S, then one of them will arrive node $z + 1$ at time $6B$ or later, and hence it will miss its deadline. Thus, we may assume that no two enforcer messages are transmitted through the same path in S [5, 6]. In our case, notice if two enforcers are transmitted through the same path, one of them will arrive node 4after the deadline600. This can be generalized for each path$1 \le j \le z$, we let $I_j = \{i \mid 1 \le i \le 3z$ and M_i is transmitted through P_j in $S\}$.

Thus, for $j = 1, I_1 = \{a_1, a_3, a_9\}$ and M_1, M_3, M_9 are transmitted through P_1; for $j = 2$, $I_2 = \{a_2, a_8, a_7\}$ and M_2, M_8, M_7 are transmitted through P_2 ;for $j = 3$, $I_3 = \{a_4, a_5, a_6\}$ and M_4, M_5, M_6 are transmitted through P_3. Clearly, this is a feasible non-preemptive transmission as shown on the right of *Figure 8*. Therefore, $MRNS$ is feasible with respect to preemptive transmission.

3.1.3.1.1.Proof by Contradiction

We have shown the theorem works for one case. To show that it holds true for all cases, we provide a generic proof. We claim that $I_1, I_2, ..., I_z$ is a solution to the 3-Partition problem. If the claim is not true, then there is an index k such that $\sum_{i \in I_k} a_i > B$. In other words, I_k is the 3 set whose sum exceeds $B = 100$.

Let $X \cup \{M'\}$ be the set of messages transmitted through the path P_k in S, where $X = \{M_i \mid i \in I_k\}$ and M' is the enforcer message. Let t' be the time instant at which M' is completely transmitted to node k, and let X_1 and X_2 be a partition of X such that X_1 and X_2 contain all the messages in X that are completely transmitted to node k before and after time t', respectively [6].

Clearly, the total time taken to transmit all the messages in $X \cup \{M'\}$ is defend in the relationship:

$$t' + \left(2B + \sum_{M_i \in X_2} li\right) \ge \left(\sum_{M_i \in X_1} li + 2B\right) + \left(2B + \sum_{M_i \in X_2} li\right) = 4B + \sum_{M_i \in X} li > 5B$$

We can interpret the above inequity by dividing the terms into 3 groups. The first group denoted by $t' + \left(2B + \sum_{M_i \in X_2} li\right)$ is the time (t') the enforcer message M' spent on the first link $(0, k)$ to

completely arrive at the intermediate node k. This is summed with $2B$ (the length of M') and the total time (length) of the partition messages to completely arrive to the destination ($\sum_{M_i \in X_2} li$) on the next link (k, e). The second group denoted by $\left(\sum_{M_i \in X_1} li + 2B\right) + \left(2B + \sum_{M_i \in X_2} li\right)$ is to account for the scenario if there are partition messages that arrives to intermediate node k before the enforcer message M' does at t'. This portion spent on the first link $(0, k)$ is $\sum_{M_i \in X_1} li$ plus the M' itself. This $\left(\sum_{M_i \in X_1} li + 2B\right)$ is then summed with $\left(2B + \sum_{M_i \in X_2} li\right)$ all the time span on the next link (k, e) as previously discussed. The third group is just algebraic manipulation but is equally expressed as the total time M' spent $(2B + 2B)$ on both links $(0, k)$ and (k, e) plus the total time all other message spent denoted by $\sum_{M_i \in X} li$ [6].

In our first non-preemptive example, analysis is trivial because all partition messages always arrive after the enforcer message finish arriving node k at time t' so set X_1 is empty. This can be seen in *Figure 9*. Hence, $200 + (2B + 100) \geq (0 + 2B) + (2B + 100) = 4B + 100 > 5B$. Here, we have $X_1 = \emptyset$ and $M_1, M_3, M_9 \in X_2$ for path $P_1 = (0, 1, 4)$; $X_1 = \emptyset$ and $M_2, M_8, M_7 \in X_2$ for path $P_2 = (0, 2, 4)$; and $X_1 = \emptyset$ and $M_4, M_5, M_6 \in X_2$ for path $P_3 = (0, 3, 4)$;
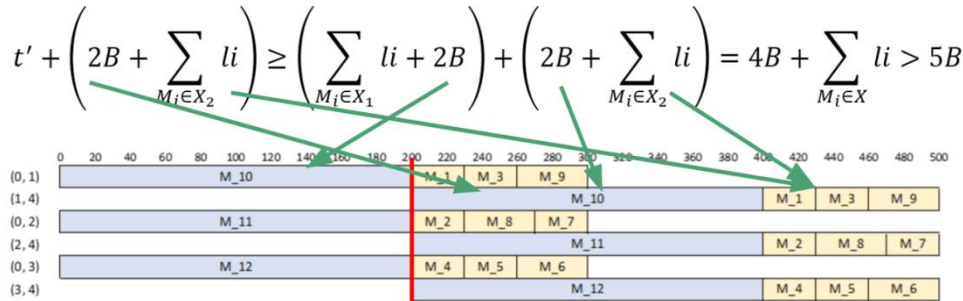

Figure 9. Non-Preemptive Transmission I (Enforcer First)

The following transmission in *Figure 10* is another non-preemptive instance where some partition messages arrive before the enforcer message finish arriving node k at time t'. We have $M_1 \in X_1$ and $M_3, M_9 \in X_2$ for path $P_1 = (0, 1, 4)$; $M_2, M_8, M_7 \in X_1$ and $X_2 = \emptyset$ for path $P_2 = (0, 2, 4)$; and $M_5, M_4 \in X_1$ and $M_6 \in X_2$ for path $P_3 = (0, 3, 4)$;
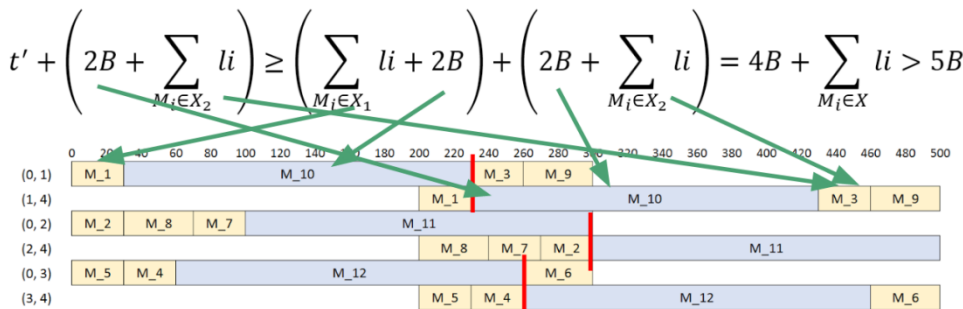

Figure 10. Non-Preemptive Transmission II (Mixed Order)

According to the definition, transmitting preemptively works as well. The following *Figure 11* shows a preemptive instance. Notice for the first link $(0, 2)$ of route P_2, M_8 was chopped and began transfer later after the enforcer M_{11} (which was also chopped several times).

$$t' + \left(2B + \sum_{M_i \in X_2} li\right) \geq \left(\sum_{M_i \in X_1} li + 2B\right) + \left(2B + \sum_{M_i \in X_2} li\right) = 4B + \sum_{M_i \in X} li > 5B$$
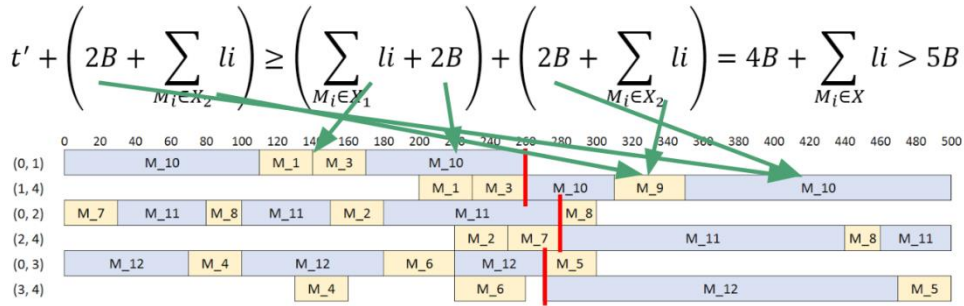

Figure 11. Preemptive Transmission I (Feasible)

These feasible transmissions are essentially best-optimal solutions. However, the whole purpose of such proof is to show that it is possible to have a non-optimal solution which exceeds $5B$. The next example *Figure 12* in take the previous in *Figure 11* and slightly delays M_6. Noticed now M_6 on link $(3, 4)$ must also be delayed because it cannot start until M_6 is fully received on node 3. This then delays the enforcer M_{12}, ultimately causing M_5 to miss the deadline (shown in red).
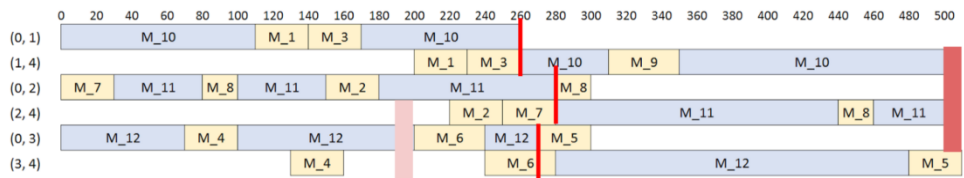

Figure 12. Preemptive Transmission II (Non-Feasible)

Thus, at least one of the messages in $X \cup \{M'\}$ has its deadline missed, contradicting our assumption that S is a feasible transmission. We showed it with a counter-example and since *MRNS* is feasible with respect to preemptive transmission if and only if *A* has a 3-Partition, if S is not feasible, therefore it is at least as hard as 3-partition [6].

3.1.3.2. Corollary 37.1

Given $MRNS = (G, M)$, where G is an arbitrary directed graph and M is a set of messages with identical origin nodes, destination nodes, release times, and deadlines, the problem of determining whether MRNS is feasible with respect to non-preemptive transmission is NP-complete [6].

Based on the previous examples, we clearly see that the Theorem 37.1 also works with non-preemptive transmission. Thus, Corollary 37.1. In fact, later in Section 4.2.2, we will show that all non-preemptive transmissions are literally preemptive transmissions.

3.2. Unidirectional Rings

In the previous section, the Theorem and Corollary have shown the message routing problem a computationally hard problem if the network is allowed to be arbitrary (Arbitrary Directed Graph). We now relax we our focus to the simpler network topology class of Unidirectional Rings in order to find the minimal makespan of the routing system. This is the simplest method in demonstrating scheduling routing in any given *MRNS*. The following *Figure 13* shows the network complexity hierarchy [5, 6].
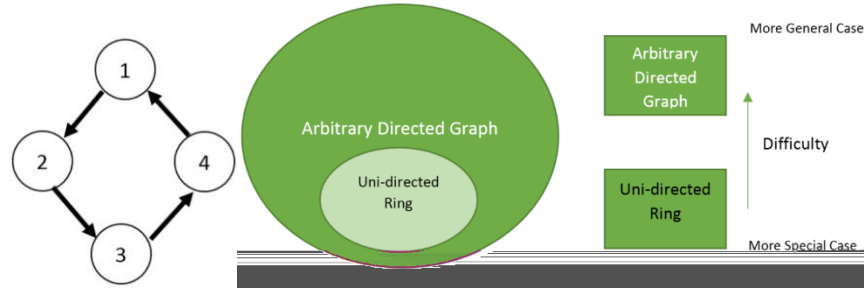
Figure 13. Unidirectional Ring Network (Left), Network Complexity Hierarchy (Right)

First, we will introduce the following additional notations and terminologies:
A simple network of "Unidirectional Ring" G is defined to be $G = (V, E)$, where $V = \{1, 2, \ldots, m\}$ for some integer $m > 1$ and $E = \{(1,2), (2,3), \ldots, (m-1, m), (m, 1)\}$ with m nodes [6].

Let S be a transmission for $MRNS = (G, M)$, where G is a Unidirectional Ring with m nodes and M is a set of n messages. We use $f(M_i, S)$ to denote the finishing time of M_i in S. For example, recall the configuration $q = (u, v, M_i, t_1, t_2)$, we have t_2 as the finishing time. In this case, $f(M_i, S) = \max_{(u, v, M_i, t_1, t_2) \in S} t_2$, the finding time of the message M_i. The makespan of S, denoted by $MS(S)$ is defined to be $\max_{1 \le i \le n} f(M_i, S)$ across all messages [6].

In our network example as shown on the left of *Figure 13*, we have $V = \{1, 2, 3, 4\}$ and $E = \{(1,2), (2,3), (3,4), (4,1)\}$ with 4 nodes.

3.2.1. Preemptive Transmission

3.2.1.1. Lemma 37.1

Let S_P be a feasible preemptive transmission for a set of messages with the same origin node, destination node, release time, and deadline. If the common origin node and destination node of all the messages are 1 and e, respectively, then we have $MS(S_p) \ge \sum_{i=1}^{n} l_i + (e - 2)(\max_{1 \le i \le n} l_i)$ [6].

For a set of messages with identical origin nodes, destination nodes, release times, and deadlines. Lemma 37.1 gives a lower bound for the makespan of any feasible transmission (both preemptive or non-preemptive). Before generalization, we will first provide an instance to better visualize the $MRNS$. Consider the following

M_i	s_i origin node	e_i destination node	l_i length	r_i release time	d_i deadline
M_1	1	4	2	0	14
M_2	1	4	3	0	14
M_3	1	4	2	0	14
M_4	1	4	1	0	14

Table 3. Message Constraints of M for the Unidirectional Ring $MRNS$

If the lemma is not true, then there is a feasible preemptive transmission S_p such that $MS(S_p) < \sum_{i=1}^{n} l_i + (e - 2)(\max_{1 \le i \le n} l_i)$. Without loss of generality, we may assume that the common release time of the messages is time 0. Let M_P be the longest message; i.e., $l_p \ge l_i$ for each $1 \le$

$i \leq n$. For each $2 \leq j \leq e$, let t_j be the time instant at which M_P is completely transmitted to node j on the edge $(j-1, j)$ in $S_p . MS(S_p) < \sum_{i=1}^{n} l_i + (e-2)(\max_{1 \leq i \leq n} l_i)$ [6].

In our example, we have $14 < (2 + 3 + 2 + 1) + (2)(3)$. In the following example, in *Figure 14*, the red lines represent the time instant that the longest message $M_P = M_2$ is completely transferred to node $2 \leq j \leq e$.
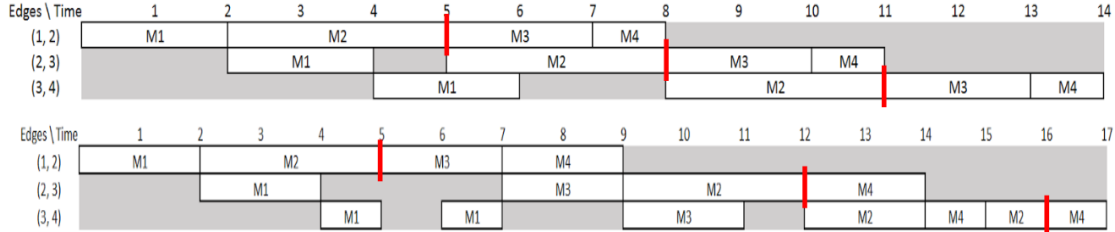

Figure 14. Time Instants t_j of S_{NP} (Top) and S_P (Bottom)

The top instance in *Figure 14* is the non-preemptive S_{NP} with $t_j = t_2 = 5$ (the time M_2 completely transmitted to node 2 on the edge $(1, 2)$), $t_j = t_3 = 8$ (to node 3 on the edge $(2, 3)$), and $t_e = t_4 = 11$ (to node 4 on the edge $(3, 4)$), the bottom instance is the preemptive S_P with $t_j = t_2 = 5$, $t_j = t_3 = 12$, and $t_e = t_4 = 16$.

It is obvious that $t_e = f(M_p, S_p)$. In our non-preemptive instance, the common destination node $e = 4$. So, the time to finish transfer the longest message M_p at t_4 is also the completion time of M_p on the last communication link (edge) [6]. Next, we will categorize the messages into two categories based on the following definition.

For each $1 \leq i \leq n$ and $2 \leq j \leq e$, let $X(i, j)$ and $Y(i, j)$ be the total amounts of M_i transmitted on the edge $(j - 1, j)$ in S_p in the time intervals $[0, t_j]$ and $[t_j, MS(S_p)]$, respectively.

This yields the two message categories $X(i, j)$ and $Y(i, j)$ for the non-preemptive transmission S_{NP} shown on the left of *Figure 15* and for the preemptive transmission S_P shown on the right. We have $X(i, j) + Y(i, j) = l_i$ for each $1 \leq i \leq n$ and $2 \leq j \leq e$ since each message is either transmitted before or after t_j.
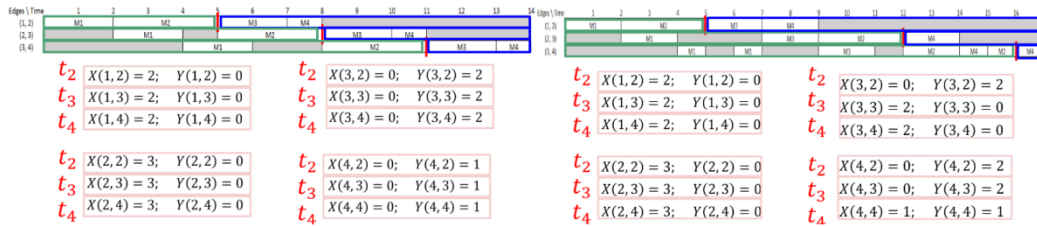

Figure 15. $X(i, j)$ and $Y(i, j)$ of S_{NP} (Left) and S_P (Right)

Next, we will categorize the messages into 3 sets. Each set has a distinct characteristic which will play in a role in determining the makespan $MS(S_p)$ of our transmission. We will define set A, B, C as follows [6]:

$$A = \{M_i \mid M_i \text{ is completely transmitted to node 2 before } t_2 \text{ in } S_p\}$$

It is clear that the messages in set A accounts for the time (length) exclusively between node 1 and node 2 on link $(1, 2)$ that finishes arriving at node 2 before M_p (the longest message) does.

$$B = \left\{ \begin{array}{c} M_i \mid M_i \text{ is completely transmitted to node 2 after } t_2 \text{ in } S_p, \text{ and} \\ M_i \text{ is completely transmitted to node } k \text{ before } t_k \text{ for some node } k, 3 \leq k \leq e \end{array} \right\}$$

The messages in set B is the ones that arrives after M_p to node 2 but later actually beats (finishes arriving earlier than) M_p on a later node (potentially any node from 3 to the destination node e).

$$C = \left\{ M_i \mid M_i \text{ is completely transmitted to node } j \text{ after } t_j \text{ in } S_p \text{ for each } 2 \leq j \leq e \right\}$$

The messages in set C are the ones that arrives after M_p does to any node ranging from node 2 to the destination node e. The total message length in all these 3 cases in fact, sum up to the entire span (makespan) of our preemptive transmission S_p.
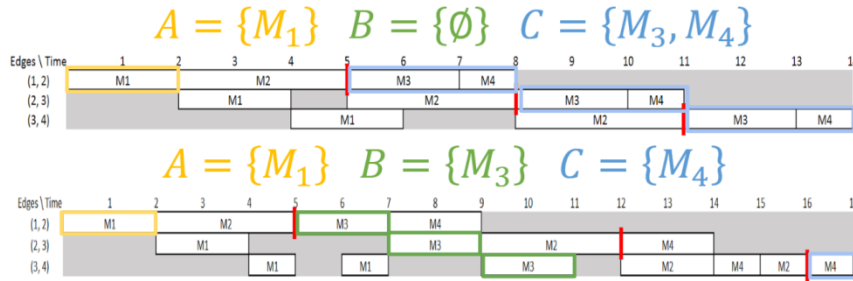

Figure 16. Sets A, B, C of S_{NP} (Top) and S_P (Bottom)

By our definitions, the sets A, B, C, and $\{M_p\}$ are pairwise disjoint, and the set $A \cup B \cup C \cup \{M_p\}$ contains all the messages. Note that M_p itself is not included in any of the sets A, B, C. With our S_{NP}, the sets for our instance is the follows with B being the empty set containing no messages. $A = \{M_1\}$, $B = \{\emptyset\}$, $C = \{M_3, M_4\}$. On the other hand, with our S_P, the sets for our instance is the follows with B being the empty set containing no messages. $A = \{M_1\}$, $B = \{M_3\}$, $C = \{M_4\}$.

We now compute a lower bound for $f(M_p, S_p)$ (or equivalently, t_e). This is the shortest/least possible time for the longest message M_p to finish transfer to the destination node. We will examine how these 3 sets of messages A, B, C form the entire makespan. So far, we have three types of messages: [5, 6]

- Each $M_i \in A$ is completely transmitted to node 2 before M_p does, and hence it delays the finishing time of M_p by at least l_i amount.
- For each $M_i \in B$, there is a node $k, 2 \leq k < e$, such that M_i is completely transmitted to node k after M_p does, but it is completely transmitted to node $k + 1$ before M_p does. Thus, it again delays the finishing time of M_p by at least l_i amount.
- Each $M_i \in C$ is completely transmitted to node $j, 2 \leq j < e$, after M_p does. However, the total amount transmitted on the edge $(e - 1, e)$ by time t_e is $X(i, e)$, and hence it delays the finishing time of M_p by at least $X(i, e)$ amount of time.

Recall that every message $M_{1 \leq i \leq n}$ is also divided into 2 groups, $X(i, j)$ and $Y(i, j)$ representing their total amounts being transmitted on the edge $(j - 1, j)$ for all edges $2 \leq j \leq e$ in S_p. $X(i, j)$

is the portion between time intervals $[0, t_j]$ and $Y(i,j)$ between $[t_j, MS(S_p)]$. In other words, $X(i,j)$ and $Y(i,j)$ arrives before and after M_p does respectively.

The properties above for $M_i \in A$ and $M_i \in B$ are self-explanatory. For $M_i \in C$, we this property so the entire set C consists of 2 parts of $\sum_{M_i \in C} X(i,j) \cup \sum_{M_i \in C} Y(i,j)$. Moreover, it is specifically used to account for the case of $X(i,e)$ when M_p is preemptively chopped in the last edge $(e-1, e)$. Hence, there exists some message $M_i \in C$ that has a portion being transferred before M_p finishes at t_e. This is the last portion to be included in $f(M_p, S_p)$ as shown in By referring to *Figure 30*, we see that our $B = \{\emptyset\}$. In our example 1, $X(1,4) = 2, X(2,4) = 3, X(3,4) = 0, X(3,4) = 0$ and only M_3 and M_4 are in C (since by definition, they finished arriving after time t_e). Therefore, $\sum_{M_i \in C} X(i,4) = 0$.

3.2.1.1.1. Direct Proof

Putting it altogether, we are able to directly proof the Lemma 37.1. We will also show it works for both our S_{NP} and S_N instances. From the above we see the finishing time of M_p is essentially t_e, which is greater than or equals the sum of the following three parts:

1. Total span (delay) of the messages in set A (denoted by $\sum_{M_i \in A} l_i$)
2. Total time to send M_p through number of edges (from node 1 to e) denoted by $(e-1)l_p$
3. Total span (delay) of the messages in set B (denoted by $\sum_{M_i \in B} l_i$)
4. Total time of the messages in set C that has part of it transferred before M_p finishes at t_e on edge $(e-1, e)$.

$$f(M_p, S_p) = t_e \geq \sum_{M_i \in A} l_i + (e-1)l_p + \sum_{M_i \in B} l_i + \sum_{M_i \in C} X(i,e) \tag{1}$$

Now, we know the makespan $MS(S_p)$ has to be the finishing time of M_p ($f(M_p, S_p)$) plus the time to finish the rest of the messages that complete after t_e ($\sum_{M_i \in C} Y(i,e)$). So, we have:

$$MS(S_p) = f(M_p, S_p) + \sum_{M_i \in C} Y(i,e) \tag{2}$$

Substituting $f(M_p, S_p)$ with $\sum_{M_i \in A} l_i + (e-1)l_p + \sum_{M_i \in B} l_i + \sum_{M_i \in C} X(i,e)$,

$$\geq \left(\sum_{M_i \in A} l_i + (e-1)l_p + \sum_{M_i \in B} l_i + \sum_{M_i \in C} X(i,e) \right) + \sum_{M_i \in C} Y(i,e) \tag{3}$$

Combining the two parts of set C (parts partially transferred before t_e during preemption and the parts after t_e) since $\sum_{M_i \in C} X(i,e) + \sum_{M_i \in C} Y(i,e) = \sum_{M_i \in C} l_i$. (Recall $X(i,j) + Y(i,j) = l_i$)

$$= \sum_{M_i \in A} l_i + (e-1)l_p + \sum_{M_i \in B} l_i + \sum_{M_i \in C} l_i \tag{4}$$

Factor out a l_p from $(e-1)l_p$:

$$= \sum_{M_i \in A} l_i + (e-2)l_p + l_p + \sum_{M_i \in B} l_i + \sum_{M_i \in C} l_i \tag{5}$$

By definition, the pairwise disjoint sets of A, B, C and M_p form the set of all messages ($A \cup B \cup C \cup \{M_p\} = \sum_{i=1}^{n} l_i$). Combining A, B, C,

$$= \sum_{i=1}^{n} l_i + (e - 2)l_p \tag{6}$$

We know that the message of the longest length can be expressed as $M_P = l_p$ which is the maximum of length l_i of all message $1 \le i \le n$

$$= \sum_{i=1}^{n} l_i + (e - 2)\left(\max_{i=1}^{n} l_i\right) \tag{7}$$

Finally, we obtain what we proposed in the Lemma. ∎ [6]

Figure17 illustrates the non-preemptive instance S_{NP} on the left and the preemptive instance S_P on the right. Both visually demonstrate each section of A, B, C in the above proof.
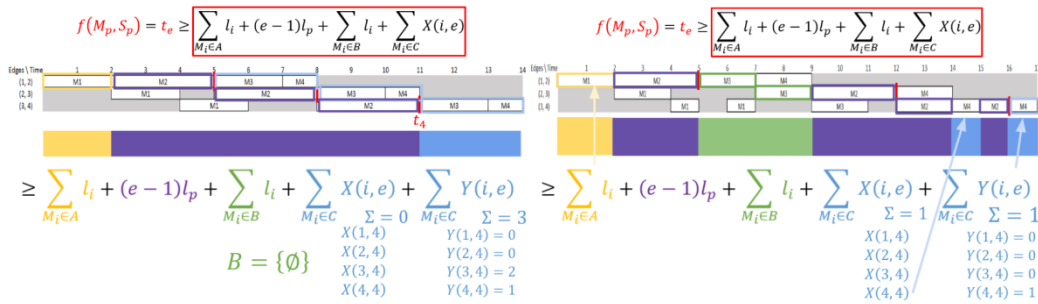

Figure 17. Visual Representation of Proof for S_{NP} (Left) and S_P (Right)

3.2.2. Non-Preemptive Transmission

We have previously proved Lemma 37.1 for a particular case of both non-preemptive and preemptive transmission. In fact, all non-preemptive transmissions are literally preemptive transmissions. It is just a special case by choosing not to utilize the preemptive functionality. Non-preemptive transmissions are a subset of preemptive transmissions. The following complexity hierarchy in *Figure 18* shows the relationship between preemptive/non-preemptive.
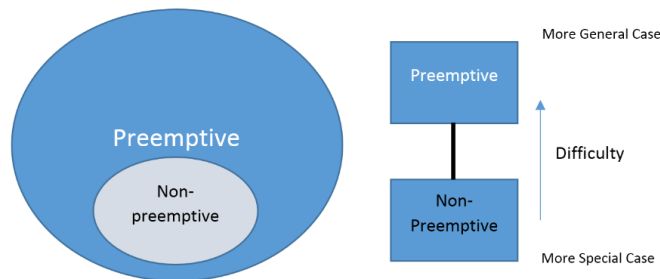

Figure 18. Non-Preemptive/Preemptive Complexity Hierarchy

This is just the rule of the general computational complexity. We have preemptive transmissions as the more general case with fewer constraints (such that each transmission must not finish transferring completely before it can be transferred elsewhere on another communication link

(link)). Intuitively, we claim that: If the more general case is easy, then the special case is easy; if the more special case is hard, then the general case is hard. This brings our next lemma:

3.2.2.1. Lemma 37.2

Let S_{NP} be a feasible non-preemptive transmission for a set of messages with the same origin node, destination node, release time, and deadline. If the common origin node and destination node of all the messages are 1 and e, respectively, then we have $MS(S_{NP}) \geq \sum_{i=1}^{n} l_i + (e-2)(max_{1 \leq i \leq n} l_i)$. [6]

This is obvious and the formal proof is as follows:

3.2.2.1.1. Direct Proof

Let S_1 be a feasible non-preemptive transmission for the set of messages such that it has the minimum makespan among all feasible non-preemptive transmissions. Let S_2 be a feasible preemptive transmission for the set of messages such that it has the minimum makespan among all feasible preemptive transmissions [6].

Since $MS(S_{NP}) \geq MS(S_1) \geq MS(S_2)$, the lemma follows immediately from Lemma 37.1.

Figure 19. Visualization of Proof for Lemma 37.2

We can clearly see in *Figure 19* that by constructing S_1 and S_2, we intermediately show that the non-preemptive transmission (our claim S_{NP}) is at least as efficient (at least as long in total transmission length makespan $MS(S_{NP})$) as the non-preemptive transmission S_1 (its makespan $MS(S_1)$) which is at least as efficient as those of the preemptive transmission S_2 (its makespan $MS(S_2)$) [6]. Therefore, we have showed:

$$MS(S_{NP}) \geq MS(S_1) \geq MS(S_2) = MS(S_P) \geq \sum_{i=1}^{n} l_i + (e-2)(max_{1 \leq i \leq n} l_i)$$

Which holds true for both preemptive and non-preemptive cases $MS(S_P) \geq \sum_{i=1}^{n} l_i + (e-2)(max_{1 \leq i \leq n} l_i)$ and $MS(S_{NP}) \geq \sum_{i=1}^{n} l_i + (e-2)(max_{1 \leq i \leq n} l_i)$ respectively for our Lemma 37.1 and Lemma 37.2 [6].

4 . KINESTHETIC LEARNING ACTIVITY MODEL

Pereducation theory, one of the most challenging aspect lays in the theory of learning. Learning theories are the conceptual frameworks in which knowledge is absorbed. The primary learning technique in most academic disciplines is the traditional learning that has been followed for ages. This typical learning technique places students in a passive rather than an active role. Traditional learning encourages one way communication and therefore, the lecture must make a conscious effort to become aware of the students' understanding without verbal feedback. Furthermore, traditional learning requires a considerable amount of time outside of the classroom to enable a long-term retention of the content [8].

To overcome the shortcomings of traditional learning and to improve the performance of all types of students, we present a technique called Kinesthetic Learning Activity (KLA). This interactive approach was first presented at California State Polytechnic University, Pomona(Cal Poly Pomona) [8]. Throughout the years of KLA implementation, research has shown substantial positive feedback from students as well as its effectiveness proven by the quizzes conducted directly after. Studies also show that students prefer a "hands on" or "learning by doing" approach to build understandings [8]. Below is the our proposed KLA specifically designed for our Real-Time Network Message Routing research.

4.1. Overview of the Proposed KLA Approach

In our KLA, we want the students to actively engage in the material directly following our presentation to provide best learning outcome. While the students may or may not comprehend the concepts of message routing over a $MRNS$, by using a "quiz" approach with hands-on, our KLA will stimulate the student's creative thinking abilities. We have developed a hands-on crafting activity using KitKat chocolate bars to simulate what the message routing configuration would be like under various preemptive/non-preemptive models.

The following *Figure 20* shows the presentation slide of the KLA introduction on left. On right is the slide detailing the assigned main activity.

Figure 20. Presentation Slides

4.1.1. Learning Goals

This study will examine the consequences of subjecting kinesthetic learning techniques on students enrolled in an upper division Computer Science course, *CS331 Design and Analysis of Algorithms*, at Cal Poly Pomona for the Summer 2017 academic quarter. Those results will be compared with the effectiveness of traditional instructional methods. The objective of this study is to provide evidence that kinesthetic learning activities supplement the learning and comprehension of course materials. Our hypothesis is that KLAs, when used to complement traditional classroom instructional methods will increase the amount of material absorbed by the students. We suggest at the end of this exercise, students will be able to experience the practical aspect of constructing a feasible transmission (whether it be preemptive or non-preemptive) given a set of messages with certain constraints. This research study has been reviewed and approved by the Cal Poly Pomona Institutional Review Board (IRB): protocol#17-116. It is in compliance with federal and state regulations as well as the Cal Poly Pomona policies regarding the protection of the human subjects used in the research.

4.2. KLA Instructions and Topic Information

4.2.1. KLA Execution Instructions

Students were given one pre-test to gauge the prior knowledge. The class will then be divided in halves. Each half will alternately be taught using the traditional method and the kinesthetic method over two class sessions by a single instructor. Following each lecture, the class was given a post-test to evaluate the level of material comprehend. Students were also asked to complete a survey afterin effort to gather opinions on the effectiveness of the KLA.

Figure 21.

During the KLA, each student was supplied a pack of 1.5-Ounce Kit Kat Bars (Milk Chocolate, but flavors may vary). Each pack consists of 4 finger bars. In addition, each participant would receive their own plastic knife, and edge/time table worksheet. The activity provides students a $MRNS$ with a diagram of an arbitrary directed graph G and a set of 4 messages $M = \{M_1, M_2, M_3, M_4\}$. Students were instructed to construct a feasible non-preemptive transmission satisfying the network and message constraints table. They were to simulate each message-span with the provided KitKat bars by crafting them into the appropriate length. Moreover, they were to place the crafted message/KitKat over the correct edge/time slot on the worksheet. Students have 15 minutes to complete the exercise. To add competitiveness, we added an additional constraint that disqualifies the students if they over-chop the message length into segments. In other words, we do not allow "gluing" pieces together. *Figure 21* shows the instructions slide on the left. On the right is the solution slide.

4.3. Evaluation of KLA Results and Analysis

The pre/post-test consists of 2 questions where Question 1 has part a (total 12 points) and b (total 3 points). The test provides a feasible $MRNS$ transmission and asks the students to fill in the blanks with 4 given available options. Question 2 is True or False (total 5 points). The testhas a total possible score of 20 points shown as follows:

4.3.1. Pre/Post Evaluation Test

M_i	s_i origin node	e_i destination node	l_i length	r_i release time	d_i deadline
M_1	1	3	4	1	6
M_2	2	5	1	2	5
M_3	1	4	2	0	6
M_4	3	4	3	0	4

Consider the above feasible $MRNS$ Routing Configuration.

1a. Fill in each blue blank with one of the following choices (M_1, M_2, M_3, M_4, or N/A) (2 points per blue blank)

1b.   Fill in each green blank with one of the following choices (M_1, M_2, M_3, M_4, or N/A)   (1 point each green blank)

Edges \ Time	1	2	3	4	5	6
(1, 2)	?		?	?		
(1, 3)		M1			M1	?
(2, 4)			?		M3	
(3, 4)		M4		?		
(4, 5)		?	?		?	

2.   True/False (1 Point Each)
   a) By changing M_4's deadline to 5 will cause this $MRNS$ to be non-feasible
   b) By changing M_2's release time to 3 will cause this $MRNS$ to be non-feasible
   c) By changing M_1's origin node to 2, this $MRNS$ will remain feasible
   d) By changing M_1's length to 3 and deadline to 4, this $MRNS$ will remain feasible
   e) This $MRNS$ is still feasible if the transmission becomes non-preemptive

The solutions of Question 1a & 1b is shown in *Figure 22* and the solution of Question 2 is a) False b) False c) False d) False e) False.
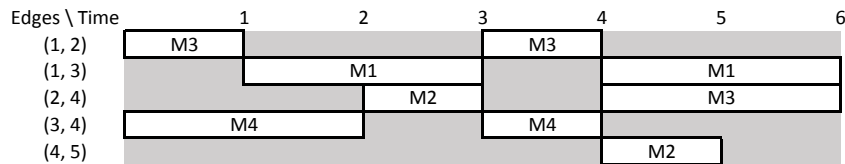
Edges \ Time	1	2	3	4	5	6
(1, 2)	M3			M3		
(1, 3)		M1			M1	
(2, 4)			M2		M3	
(3, 4)	M4			M4		
(4, 5)					M2	

Figure 22. Solutions to Pre/Post-Test Question 1a & 1b.

4.4. KLA Experimental Results

The results of our KLA turned out well. Students all had a positive response to the activity. The scores from the tests are anonymously recorded in *Figure 23*showingthe pre-test and post-test scores, the mean, median, max, min points and percentages, along with the total number points possible. Overall, although the KLA half of the class had a lower score throughout, they improved more significantly than the traditional teaching half.

Results	Traditional		KLA	
	Pre-test	Post-test	Pre-test	Post-test
	11	16	5	15
	13	10	7	13
	15	19	6	15
	12	11	11	11
	11	18	13	13
	7	11	14	14
	18	16	6	11
	8	14	5	18
	15	8	12	14
	10	17	8	6
	1	16	10	10
	12	16	8	16
	5	18	5	17
	4	17	15	9
	14	8	6	14
	13	10		
	13	11		
Total Students	17	17	15	15
Mean	10.70 (53.52%)	13.88 (69.41%)	8.73 (43.67%)	13.06 (65.33%)
Median	12 (60.00%)	16 (80.00%)	8 (40.00%)	14 (70.00%)
Max	18 (90.00%)	19 (95.00%)	15 (75.00%)	18 (90.00%)
Min	1 (5.00%)	8 (40.00%)	5 (25.00%)	6 (30.00%)
Out of	20	20	20	20
Score Improvement	15.88		21.67	

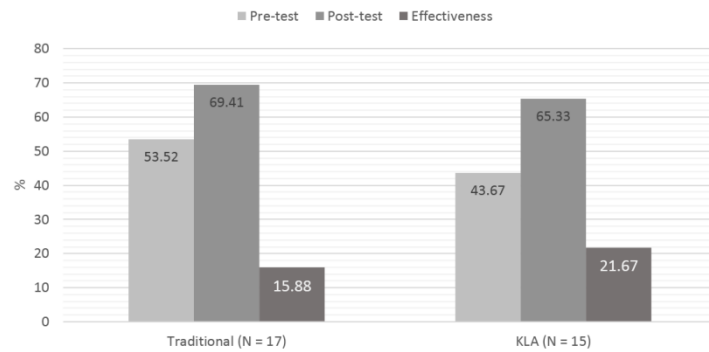Figure 23. KLA Pre/Post-test Results

Figure 24. % of Average Test Scores & Learning Effectiveness

Figure 24 shows the difference in the pre-test and post-test results and separates the traditional vs KLA groups. The improvement in the average score for the group that learned via the traditional method was 15.88%, while the KLA group saw their average score jump by 21.67%.

4.4.1. KLA Survey Feedback

At the end of our KLA trial, the total of 29 students completed the survey consisting the following questions:

- **Question 1**: My overall evaluation of the kinesthetic learning activity (KLA) is (circle one): *Excellent, Good, Fair*, and *Poor*.
- **Question 2**: Rate the following questions on a scale of *Strongly Agree* to *Strongly Disagree*:
  a)   I enjoyed the KLA experience.
  b)   The kinesthetic activities were easy to understand.
  c)   The kinesthetic activities were easy to follow.
  d)   KLA is effective in helping me learn concepts.
  e)   KLA helped me be more engaged in learning process.
  f)   I would recommend KLA as an alternative way of teaching

The responses from Question 2 is shown in the bottom of *Figure 25* indicating most the students had a favorable experience with the KLAs. There were very few responses of Disagree or Strongly Disagree. This result is consistent with Question 1. At last, all students participated and enjoyed eating the KitKats. It was a good attempt considering it was the first KLA involving food. We have also discovered many improvements that we can make to better the learning outcome.
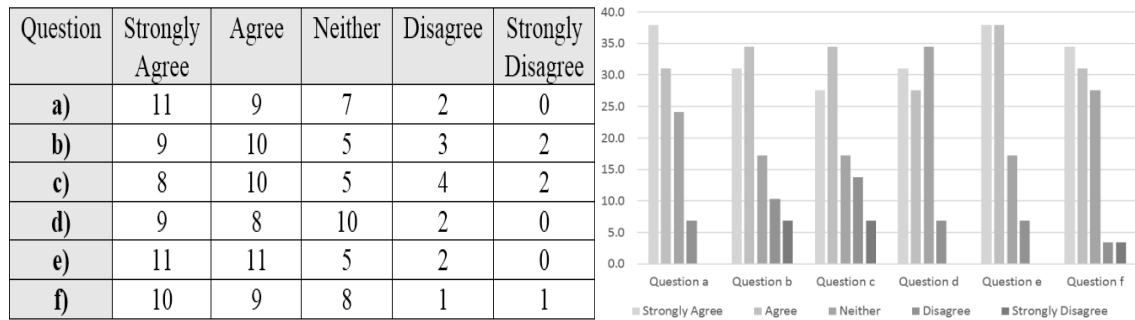

Selection	# of respondents
Excellent	7
Good	13
Fair	9
Poor	0

Question	Strongly Agree	Agree	Neither	Disagree	Strongly Disagree
a)	11	9	7	2	0
b)	9	10	5	3	2
c)	8	10	5	4	2
d)	9	8	10	2	0
e)	11	11	5	2	0
f)	10	9	8	1	1

Figure 25. Results of Survey Question 1 (Top) Question 2 (Bottom)

Figure 26. Students Executing KLA in Action

5. CONCLUSIONS AND FUTURE WORK

In our survey of routing real-time messages, we have considered the problem of how to construct a feasible transmission on a given *MRNS* if one exists. In the case where the *MRNS* does not have a feasible transmission, we identified the specific constraints that caused it to miss the deadline. In addition, we analyzed and proved the computational complexity of an Arbitrary Directed Graph network to be NP-complete even when all four parameters are fixed. We then considered a simpler network which is a special case of the Arbitrary Directed Graph called a Unidirectional Ring. The same complexity results hold for preemptive transmission.

Lastly, to better demonstrate this *MRNS*, we proposed a Kinesthetic Learning Approach way to model the scheduling with KitKat chocolate bars. This KLA method promotes student interactivity and improves student learning by engaging a fundamental and easily conceived learning style. As previous research suggest, courses on analysis of algorithms are ideally suited for KLAs due to the complexity of the subject matter and the ability to adapt algorithm steps into physical actions. Our KLA results a positive feedback from the students. In addition, our case study shows an improvement in pre-test/post-test scores to support its effectiveness. For future work, we plan to expand our research to more offline routing variations as well as online routing of variable-length messages with more variations in network complexity such as out-tree, in-tree, bidirectional tree, and bidirectional ring. We also plan to improve our KLA model by providing better instructions through video examples for better participating results. We will compare the experimental results to quantitatively analysis on its effectiveness.

REFERENCES

[1] R. Koch, B. Peis, M. Skutella and A. Wiese, "Real-Time Message Routing and Scheduling", TU Berlin, Institut f¨ur Mathematik, Straße des 17. Juni 136, 10623 Berlin, Germany,, 2017.

[2] U. Shaw and B. Sharma, "A Survey Paper on Voice over Internet Protocol (VOIP)", International Journal of Computer Applications, vol. 139, no. 2, pp. 16-22, 2016.

[3] D. Hemmendinger, "The ACM and IEEE-CS guidelines for undergraduate CS education", Communications of the ACM, vol. 50, no. 5, p. 46, 2007.

[4] P. Mueller, "New Algorithm Significantly Boosts Routing Efficiency of Networks", Phys.org, 2008. [Online]. Available: https://phys.org/news/2008-08-algorithm-significantly-boosts-routing-efficiency.html. [Accessed: 30- May- 2017].

[5] J. Leung, T. Tam and G. Young, "On-Line Routing of Real-Time Messages", Journal of Parallel and Distributed Computing, vol. 34, no. 2, pp. 211-217, 1996.

[6] J. Leung, Handbook of Scheduling, 1st ed. London: Chapman & Hall/CRC, 2004, pp. 795-826.

[7] S. Joosten, "Relaxations of the 3-partition problem", Department of Applied Mathematics, University of Twente, Enschede, The Netherlands, 2011.

[8] Alraddady, Sara, Danny Luong, and Gilbert Young. 2014. "A Study of Kinesthetic Learning Activities Effectiveness in Teaching Computer Algorithms." The 2014 International Conference on Frontiers in Education: Computer Science & Computer Engineering. 400-406

AUTHORS

Paul is a Graduate Student of Computer Science at Cal Poly Pomona. His research interests include assistive technology for the disabled, software accessibility testing, web and mobile applications, and CS Education, etc. He will be pursuing his Ph.D. in CS and wishes to one day become a CS Professor.

Dr. Sun Assistant Professor of Computer Science at Cal Poly Pomona. His research interests include software engineering, cloud and mobile computing, etc. Sun leads the research and development for numerous start-up companies. He has a solid background in both the industry working at Amazon as well as academia working as a Post-Doc research associate at Vanderbilt University.
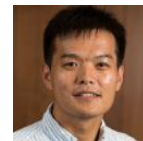
Dr. Young Professor of Computer Science at Cal Poly Pomona. His research interests include parallel and distributed computing, computer networks, algorithm design, scheduling, combinatorial optimization, etc. Young is the creator of the KLA and has received the College of Science Distinguished Teaching Award for his innovative and continuous research on this alternative CS teaching approach.