Collective Communications on a Beowulf Cluster: Companion Document

Wi Bing Tan and Peter Strazdins,
Department of Computer Science,
Australian National University

1. Introduction

This paper is a supplement to the paper *The Analysis and Optimization of Collective Communications on a Beowulf Cluster* [1], containing diagrams illustrating the All-Gather, All-Reduce and Reduce-Scatter communication patterns and generated by a communication pattern simulator.

It should be read with this paper; it is not intended as a read-alone document.

This paper is organized as follows. Section 2 gives a diagram of the Bunyip configuration. Section 3 gives diagram illustrating how collective communication operations for various algorithms. Diagrams generated the Simulator Tool are given in Section 6.

2. The Bunyip Cluster

Figure 1 shows the network topology of Bunyip.

3. Algorithm Description and Performance Models

3.1. Performance model for Point-to-point Messages

3.2. Operations

3.3. Communication Patterns

3.3.1 Bi-Directional Exchange

Figure 2 illustrates the Bi-Directional Exchange pattern.

3.3.2 Recursive-Halving Recursive-Doubling

In the case of the All-Reduce and All-Gather operations Recursive-Halving Recursive-Doubling works by simultaneous exchange of data between nodes as shown in the Figure 3. Reduce-Gather is illustrated by the top half of this Figure.

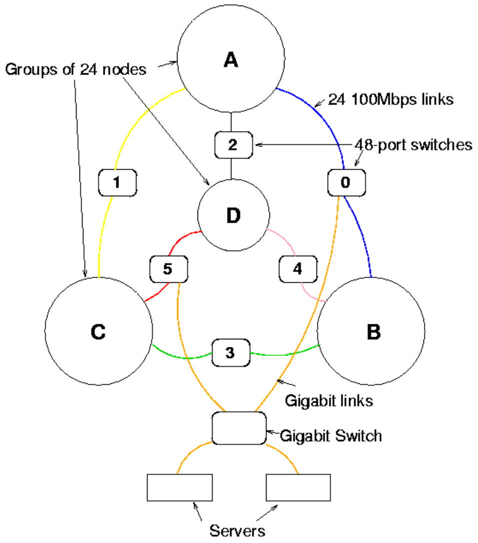

Figure 1. Network Topology of Bunyip

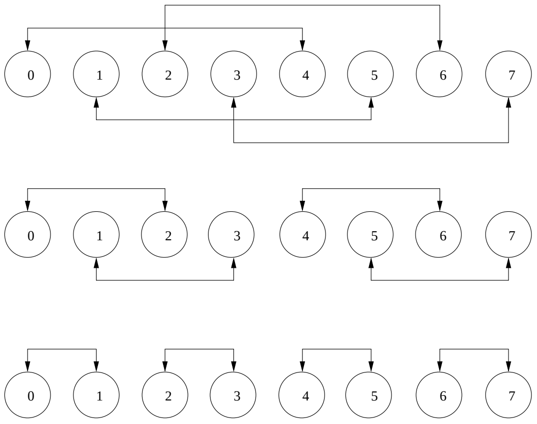

Figure 2. Bi-Directional Exchange

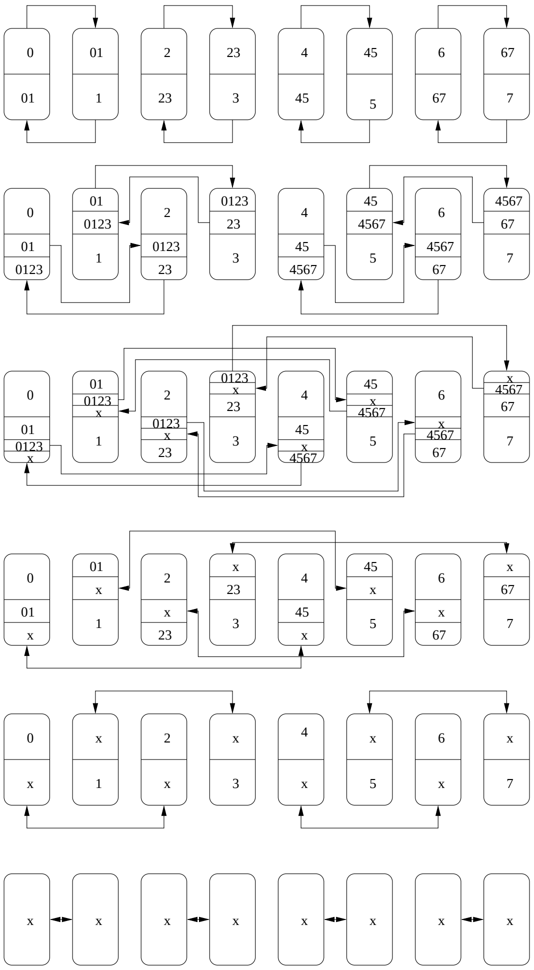
0 01 | 01 1 | 2 23 | 23 3 | 4 45 | 45 5 | 6 67 | 67 7
0 01 0123 | 01 0123 1 | 2 0123 23 | 0123 23 3 | 4 45 4567 | 45 4567 5 | 6 4567 67 | 4567 67 7
0 01 0123 x | 01 0123 x 1 | 2 0123 x 23 | 0123 x 23 3 | 4 45 x 4567 | 45 x 4567 5 | 6 x 4567 67 | x 4567 67 7
0 01 x | 01 x 1 | 2 x 23 | x 23 3 | 4 45 x | 45 x 5 | 6 x 67 | x 67 7
0 x | x 1 | 2 x | x 3 | 4 x | x 5 | 6 x | x 7
x x x x x x x x

Figure 3. Recursive-Halving Recursive-Doubling

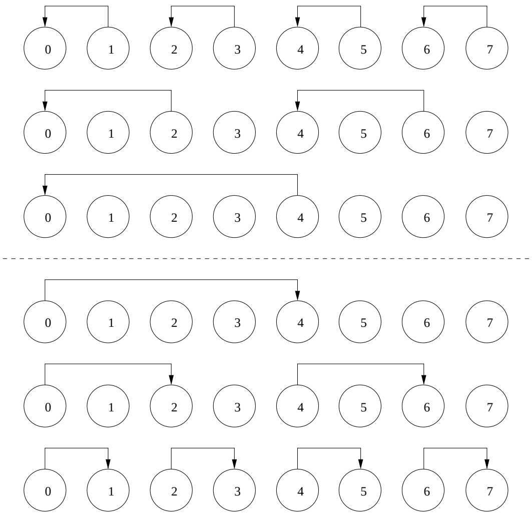

Figure 4. Fan-in Fan-out

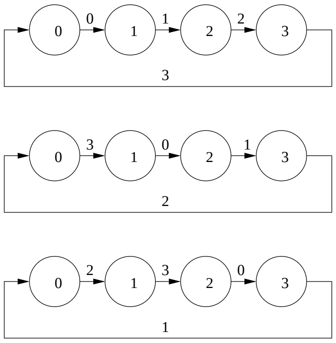

Figure 5. Ring

3.3.3 Fan-in Fan-out

Figure 4 illustrates the Fan-in Fan-out pattern for All-Gather and Reduce-Scatter.

3.3.4 Ring

Figure 5 illustrates the Ring pattern.

3.3.5 Repeated Binary Tree

Figure 6 illustrates the Repeated Binary Tree pattern.

3.3.6 Pipeline

Figure 7 illustrates the Repeated Pipeline pattern.

3.3.7 Full Fan-in

Figure 8 illustrates the Full Fan-in pattern.

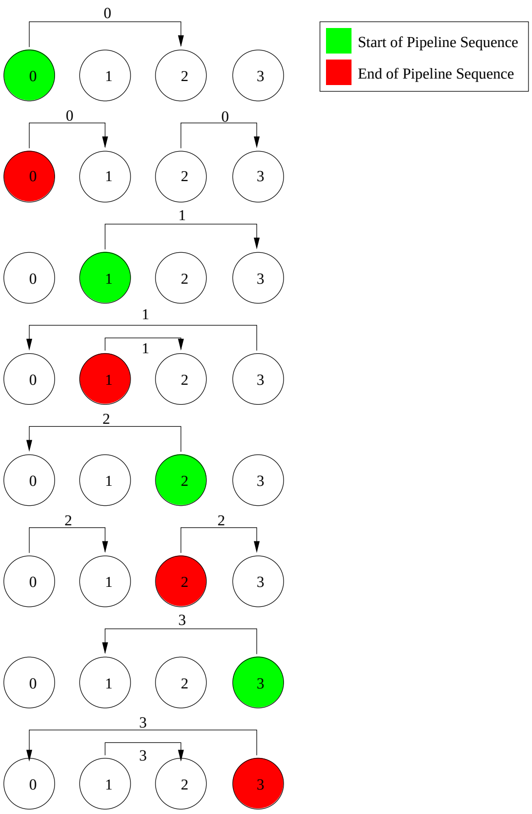

Figure 6. Repeated Binary Tree

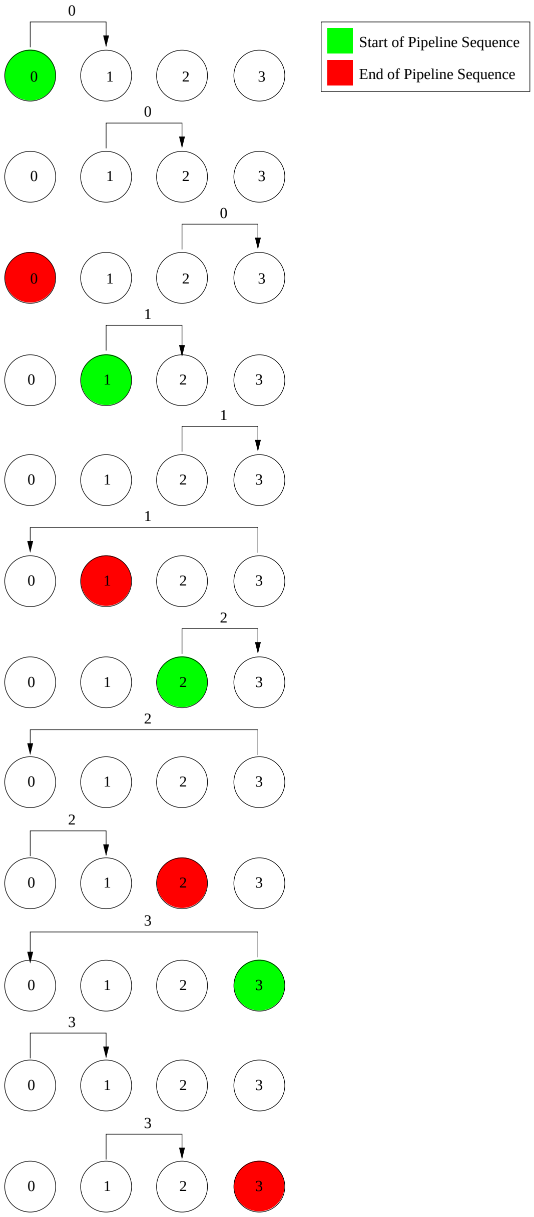

Figure 7. Pipeline

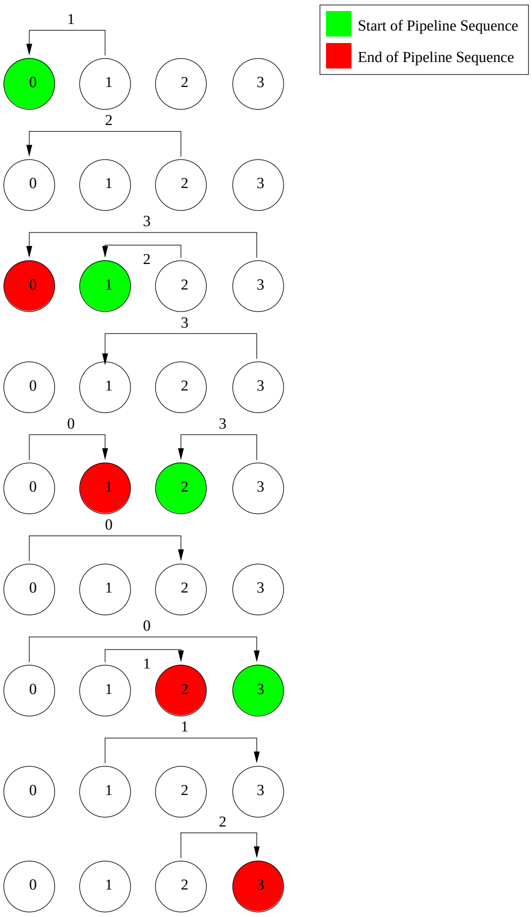

Figure 8. Full Fan-in

4. Results

5. Inter-Group Communication Patterns

6. Simulator

6.1. Example

Figures 9 and 10 shows the visualization of the Full-Fan-In pattern, generated by the Simulator Programmer and the Diagramming Tool, respectively. Figure 9 was used to derive the performance model for this pattern.

Figure 11 show the simulator visualization of the Repeated Binary Tree pattern for All-Gather, indicating the overlap between sub-operations. This was used to derive the pattern's performance model.

7. Conclusions

References

[1] W. B. Tan and P. Strazdins. The analysis and Optimization of Collective Communications on a Beowulf Cluster. In *Proceedings of ICPADS'02: 2002 International Conference on Parallel and Distributed Systems*, Dec. 2002.

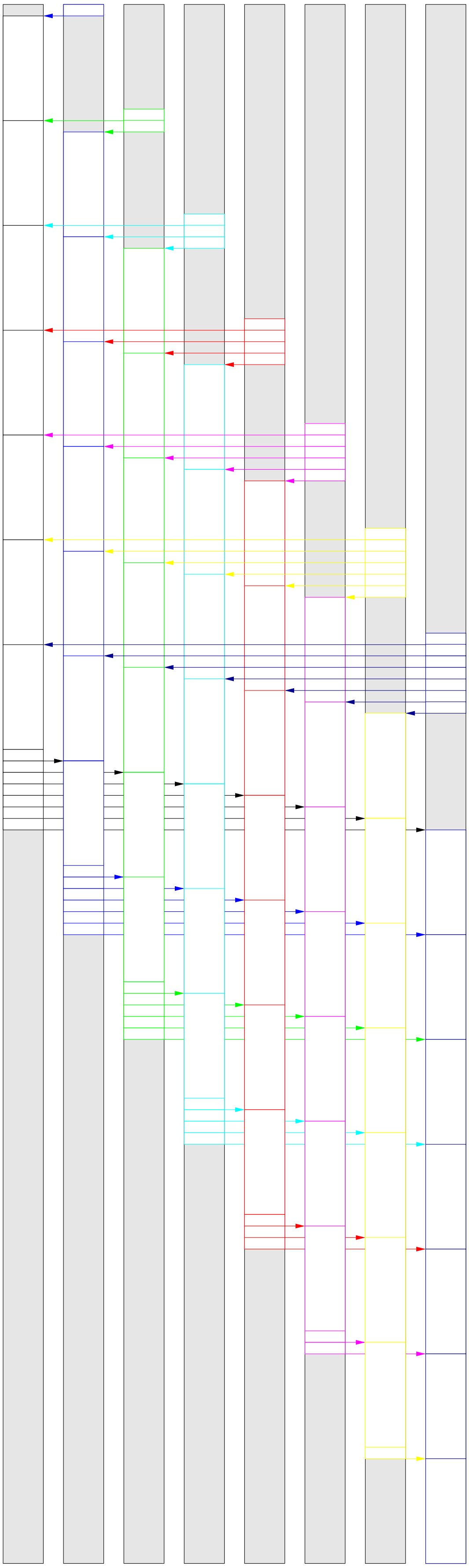

Figure 9. Time Chart for Full Fan-in (Simulator)

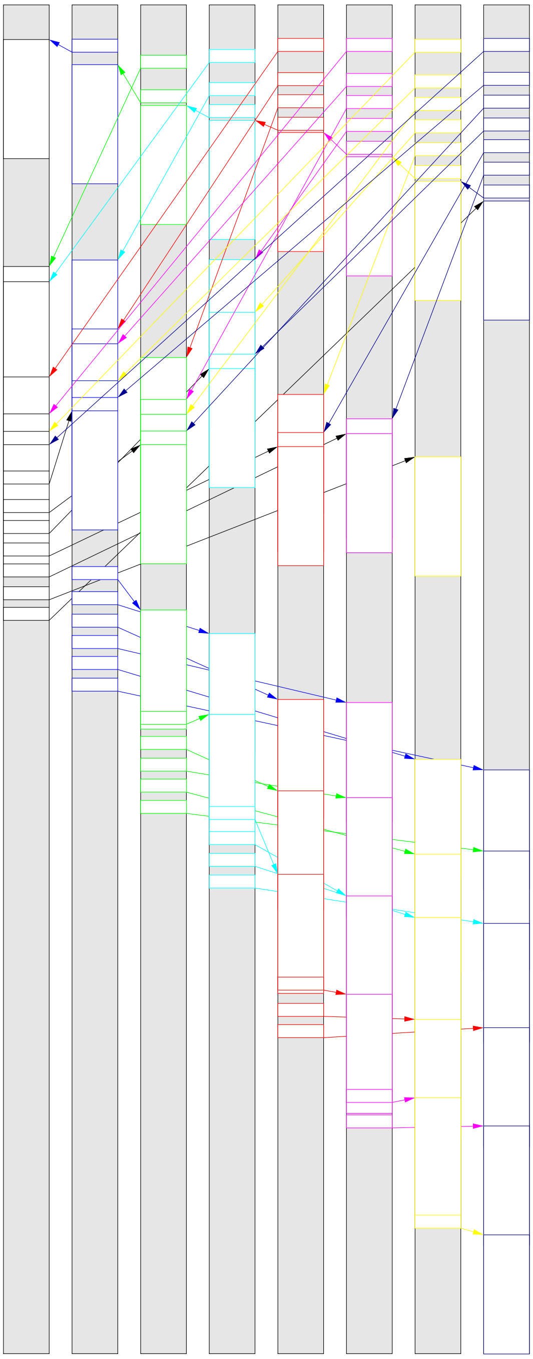

Figure 10. Measured Time Chart for Full Fan-in (Diagramming Tool)