Chapter 8

Future Work in Isochronets

8.1 Introduction

Isochronets present several open challenges. Some important open problems are natural extensions of initial findings reported in this thesis. They were presented at the end of previous chapters. This chapter will describe other important future directions in Isochronets.

8.2 Large Isochronets

8.2.1 The Problem

How to design Isochronets with large number of nodes? If the number of nodes in the network is very large, it may be necessary to partition the clock cycle into many bands. As a consequence, each band may become very small. Small bands may under-use network resources and increase the time waiting for bands. For example, if the network contains 1,000 nodes and the network topology only allows two trees per band, the mean band duration with a 125 μs cycle is only 125 ns. Links operating at 1 Gb/s take 177 ns to transmit an ATM cell, more time than the mean band size. Additionally, the average time waiting for a band is more than 62 μs.

8.2.2 Preliminary Solution Overview

The proposed solution is to cluster nodes and use Isochronets both within and between clusters. For example, a two level solution would use Isochronets among the nodes in the same cluster (*in-cluster Isochronet*) and between clusters (*inter-cluster Isochronet*), as depicted in Figure 8.1. An illustrative Isochronet network is shown for Cluster E. It contains one or more gateway machines to other clusters, that is, machines that contain links connected to gateways in other clusters.

The following is a typical scenario to send frames from a cluster. In Figure 8.1, let us assume that a node *n* in Cluster E needs to send a frame. If the destination also resides in Cluster E, *n* uses the in-cluster Isochronet band within Cluster E to the destination. Otherwise, *n* sends the frame to the gateway machines *g* in Cluster E. Gateway *g* parses the frame header and schedules its transmission to the destination cluster using the inter-cluster Isochronet. Once the frame reaches the final cluster, the cluster gateway parses its header and then identifies the ultimate destination machine. It schedules the frame for transmission within the in-cluster Isochronet band to the destination machine.

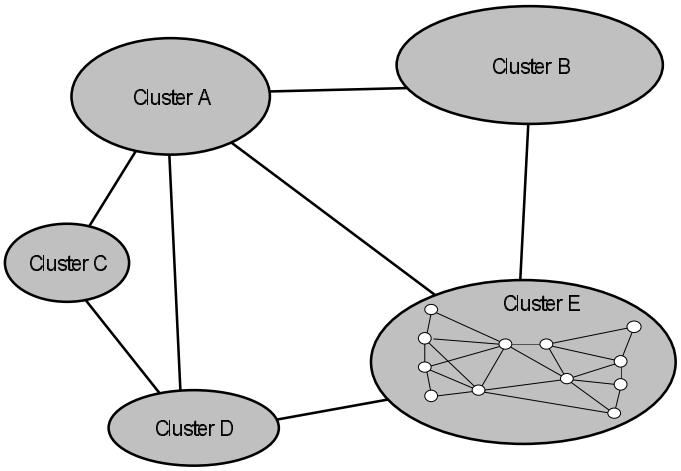

Figure 8.1: Two level hierarchical clustering of Isochronets

This architecture can be extended to multi-cluster hierarchies, where, in each level, Isochronets would run within a cluster and between the clusters.

This solution may offset many of the Isochronets advantages. For example, frame processing is necessary at gateways. The solution can be seen as a compromise between providing advantages of Isochronets on one hand and dealing with large networks on the other hand. Further study must be pursued to best implement a balanced compromise between these goals. The following are a few open issues regarding the proposed solution:

- *How to cluster the machines in order to minimize frame processing?* On the one extreme, one cluster will imply no frame processing while, on the other extreme, too many clusters will imply considerable processing at gateways.
- *How to assign and resolve addresses?* Gateways need to map addresses into clusters and machines efficiently. Nevertheless, it might be necessary to maintain addresses compatible with other networks such as, for example, Internet.
- *How to minimize multi-protocol processing?* One of the important aspects of Isochronets is the absence of multi-protocol adaptation. On the one extreme, an internal uniform frame structure could ease frame parsing at gateways. At the other extreme, multiple protocol structures may complicate gateway processing.

8.3 Reliability in Isochronets

8.3.1 The Problem

How to overcome link and node failures? Link failures in Isochronets will disrupt routing trees and thus disable some connections. Node failures will block all connections

passing through them. In such scenario, new routing trees need to be dynamically setup to overcome the disruptions.

8.3.2 Preliminary Solution Overview

Isochronets are particularly suitable for duplication techniques. One possible technique to ensure reliability is to allocate multiple trees to the same destinations. Different trees to the same destination are assigned different bands. That is, each destination would have multiple distinct routing trees assigned to different bands. Also, the routing trees would be allocated so that a pre-defined number of link (or node) failures would not disrupt all the routing trees to any destination. In this fashion, the network would still operate to all destinations by using the band that contains a valid routing tree.

Figure 8.2 illustrates two non-interfering routing trees to the same destination. In a more general case, some of the links may be shared. The idea is that, if each tree is assigned to a different band, the destination is reachable using both bands. If one of the links or nodes fails, it may still be possible to reach the destination using one of the bands.

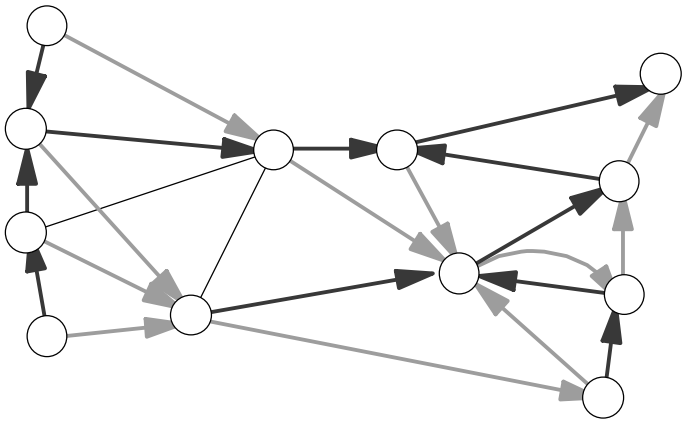

Figure 8.2: Two non interfering routing trees to a single destination

The goal is to find allocations such even if *k* links or *k* nodes fail, there is still a path from all sources to a given destination in some band.

Bands will contain two kinds of trees. The original trees to each destination and backup trees that may or not be used. If a band consists only of backup trees, its size may be shrunk to 0. When failures occur, it can progressively increase to overcome the failure.

The following are a few open issues regarding the proposed solution:

- *How should the trees be allocated?* This problem generalizes the tree allocation problem in Chapter 4 to cover all possible *k* link or node failures.
- *Given a generic network, what are the maximum node and link failures that may be supported?* This is an indication of how reliable the topology is.
- *What is the minimal number of bands that is needed to cover all possible k link or node failures?* The more bands are necessary, the more difficult the problem of deciding band sizes for original and backup trees becomes.
- *How should the band sizes be affected by failures?* A failure in a link may direct considerable traffic to other bands.

8.4 Reconfiguration

8.4.1 The Problem

How should band and tree allocations change when routes change? Routes may change due to the introduction or removal of nodes or links. Also, it is possible to change the routes with alternative more efficient options.

8.4.2 Discussion

Upon the introduction of new nodes or links in the network, it is necessary to re-compute tree and band allocations. The ideal solution would involve incremental updates to existing allocations.

8.5 Topology Design

8.5.1 The Problem

Suppose one is to design a network that will operate according to the Isochronets architecture. *What is the most suitable topology taking into account multiple constraining issues such as cost, node locations, etc.?*

8.5.2 Discussion

Many techniques have been applied to generic optimization problems as the one proposed here. One of the difficult questions is how to define the optimization criteria that would best accommodate each scenario.