

# Cartesian Routing

## Progress to Date and Future Directions

Larry Hughes\* Omid Banyasad†

Department of Electrical and Computer Engineering  
Dalhousie University, Halifax, Nova Scotia, B3J 2X4, Canada

1 February 2001

### Abstract

Existing hierarchical routing schemes, employing distance vector and link state routing algorithms, require the exchange of routing information for the construction and maintenance of routing tables [1]. As networks increase in size, the memory requirements for the routing tables and the time taken to search the tables increase proportionally [11]. Further, as the popularity of computer networks increases, the size of the address space can become a limiting factor [9]. Although many Internet routers employ specialized caches to hold recently-used addresses, table search times can degenerate to  $O(\log(n))$  and  $O(n)$ , for ordered and unordered tables, respectively [2, 10]. Furthermore, routers become the bottleneck in high-speed optical networks since packets must be converted from the network's media (light) to the router's media (electrical).

In this paper, the authors discuss a novel packet routing technology known as **Cartesian routing** that is applicable to both small and large scale networks [7]. Cartesian routing differs from existing provider-based unicast routing in that routers maintain a minimal amount of state information. Routing tables are unnecessary since communications are topologically dependent, potentially reducing router and network overheads. Routing decisions which can take  $O(\log(n))$  to  $O(n)$  time using routing tables are reduced to  $O(1)$  time in Cartesian routing.

The paper describes the fundamental Cartesian routing algorithm, its topological and addressing requirements, and its recursive implementation for global internetworks. The paper also presents a description of a Cartesian network simulation tool, as well as a prototype Cartesian router. Finally, the paper examines potential future research areas including the design of optical Cartesian routers, improved fault tolerance, network interconnection, and some of the issues surrounding possible designs of Cartesian Internets.

---

\* Author to whom correspondence should be addressed.

† Faculty of Computer Science, Dalhousie University.

# 1 Cartesian Routing and Cartesian Networks

The fundamental principles of Cartesian routing can be illustrated using a linear routing algorithm in an one-dimensional network topology [5, 6]. In this topology, each router is associated with two ‘side’ ports (east and west), allowing it to connect to, at most, two other routers. Every router is bound to a unique address and maintains no state information other than this address.

Linear routing is achieved in the network by imposing an ordering on the routers which is based upon the unique router addresses; for example, west-to-east in ascending order (unless otherwise specified, west is always considered ‘less than’ east). When a packet is to be transmitted on the network, the transmitting router’s layer 3 determines the packet’s initial direction by examining the destination address. If this is less than the router’s address, the packet is queued for transmission on the west port, otherwise on the east port. When a packet arrives at a router, its address is compared with the router’s address: if the addresses are the same, the packet can be kept; if the packet arrived from the east (west) and is greater than (less than) the router’s address, it is discarded; otherwise the packet is forwarded out the opposite port from which it arrived.

A Cartesian network consists of a set of **collectors** and one or more **arterials**, as shown in Figure 1. Each collector is a chain of collector routers running east-west, sharing a common latitude. Collector routers have two ports (east and west) to exchange packets “horizontally”. Each collector router also has a **bottom port** which allows it to connect to a set of local hosts. Arterials exchange packets between collectors. Each arterial router, except the most northerly and the most southerly, has, at least, four ports (north, south, east and west). Arterials need not share a common longitude.

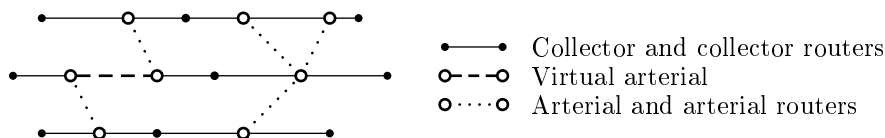


Figure 1: A Cartesian Network

In a Cartesian network, the imposed topological structure relieves each router from maintaining routing tables. Each router is bound to a unique address (for example, a latitude and longitude). Both collector and arterial routers implement the linear routing algorithm: collector routers examine latitudes while arterial routers examine longitudes.

The state information maintained is minimal: each router maintains an **Arterial Direction Indicator** (ADI) that indicates which of its latitudinal ports leads to an arterial and whether the arterial connects to the north, south, or both. A router’s ADI is updated when an arterial router sends an **Arterial This Way** (ATW) message out its collector ports or when a router detects a change in the state of its links.

An arterial router differs slightly from a collector router in that it can have multiple links leading to other arterials, for both fault tolerance (should an arterial link fail) and potential shortcutting. A **virtual arterial** is a collector that doubles as an arterial, allowing a continuous path from north-to-south, as shown in Figure 1. A detailed description of Cartesian routing can be found in [7].

## 1.1 Progress to Date

### 1.1.1 Prototype Collector Router Implementation

A prototype multiprocessor collector router has been developed for demonstration purposes [4]; the router supports three full-duplex RS-232 ports (bottom, east, and west).

Each port is controlled by an Atmel AT90LS8535 RISC microprocessor with a built-in UART and a 486 byte buffer. Each microprocessor is associated with a 1 kilobyte FIFO (to hold packets queued for output) and two tri-state octal buffers (controlling access to the FIFOs of the other microprocessors). The 128-bit address of the router is stored in each microprocessor.

When a packet starts to arrive at a port, the microprocessor compares the destination address with the router’s address to determine the output port. If the packet is not to be discarded, the microprocessor writes

the destination address and the remainder of the data packet to the FIFO of the proper port. When a FIFO receives a byte, it signals its microprocessor that outgoing data is present. The microprocessor immediately reads the data and transmits it through its RS-232 connection.

The multiple bus architecture gives the router the ability to create three isolated buses. Data flows uninterrupted if packets received on two separate ports do not have to go out the remaining third port. When a microprocessor sends data to another port's FIFO memory it cannot be interrupted by the third port's microprocessor. If the remaining processor has data intended for a port that is already in use, that port will have to temporarily store the packet until the FIFO becomes free. A port's processor writes only one data packet to a FIFO memory if another port has data destined for the same port. After one packet is written, the other processor has the opportunity to write a packet to the FIFO. This sharing process allows data received from two ports to flow out one port.

### 1.1.2 Visualization Tool

A web-based visualization tool that demonstrates the operations of a Cartesian network has been written. The tool allows the user to design a network. The network topology is validated as the network is being built; once built, the user can choose packet source(s) and destination(s) and then run the network. After the network initializes itself and begins routing packets, the user can choose new packet source(s) and destination(s), simulate link failure and recovery, and examine router status (since each router is an object that maintains statistics on its reachability as well as the number of packets handled). The URL for the visualization tool is [is2.dal.ca/~rtmurphy](http://is2.dal.ca/~rtmurphy).

## 2 Wide Area Cartesian Networks

A Cartesian network provides a straightforward topological structure that relieves routers from maintaining routing tables. However, it is unrealistic to implement a single worldwide Cartesian network, since such a network would, for example, require every packet destined to a router with the same latitude identifier as the source router's to visit all the collector routers in between. It is also necessary for such a network to have one collector for every possible latitude. These limitations suggest that implementing a single worldwide Cartesian network would be impractical. An alternative to a worldwide Cartesian network is to create a set of smaller Cartesian networks and implement a mechanism for interchanging packets between them. In this section, a multiple-layer Cartesian network is expanded using a hierarchical structure [8].

### 2.1 Topology

Multiple-layer Cartesian networks have a hierarchical structure. The highest layer of the hierarchy is a single Cartesian network. Each underlying layer consists of a set of mutually disjoint Cartesian networks (i.e., they are physically disjoint, sharing no collector or arterial router). A layer of the hierarchy is referred to as **layer-n**. The lowest layer of the hierarchy is layer-1 and the highest layer is layer-m, where 'm' is the maximum number of layers. Collector routers in layer-1 are connected to local hosts through their bottom port. Each collector router at layer-n represents a single Cartesian network at layer-(n-1), where  $1 < n \leq m$ . When a collector router R in network A at layer-n represents a Cartesian network B at layer-(n-1), it is said that network A and collector router R "encompass" network B.

In an m-layer Cartesian network, each network at layer-n,  $1 \leq n < m$ , is connected to another network at layer-(n+1) via a single **Internetwork Router (IR)**. An IR is a bridge, interchanging packets between a Cartesian network and its "immediate" encompassing network. Each IR is an arterial router with an additional port, the **top port**, that connects a network at layer-n to its encompassing network at layer-(n+1). Figure 2 illustrates an encompassing network (A) and two encompassed networks (B and C) along with two IRs.

### 2.2 Router Identification

In a multiple-layer Cartesian network, each router is bound to an **identifier**. The identifier of a router at layer-m of an m-layer Cartesian network is the same as its Cartesian address. An identifier of a router at

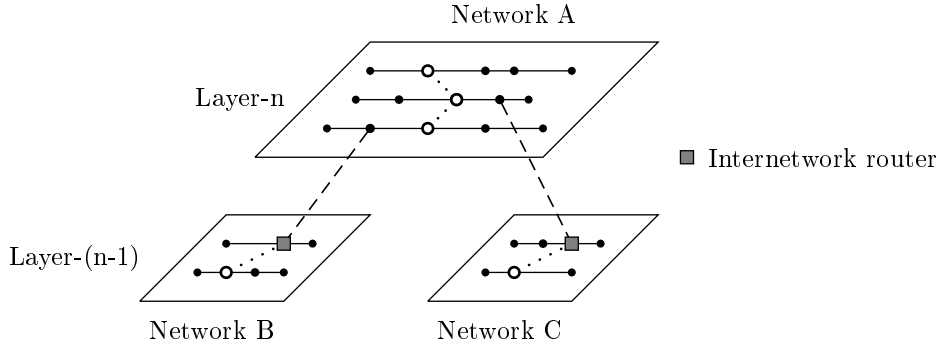


Figure 2: Network A encompasses networks B and C

the lower layers is the identifier of its immediate encompassing router, followed by the Cartesian address of the router itself, meaning that an identifier is an ordered list of Cartesian addresses. In general, in an  $m$ -layer Cartesian network, each router at layer- $n$  maintains a list of  $(m-n+1)$  ordered Cartesian addresses (where  $1 \leq n < m$ ). For example, routers at the lowest layer of the hierarchy, layer-1, which are connected to local hosts through their bottom ports, are bound to ‘ $m$ ’ ordered Cartesian addresses:  $m-1$  correspond to the identifier of the encompassing router at layer-2, and another one, the Cartesian address of the router itself, as shown in Figure 3.

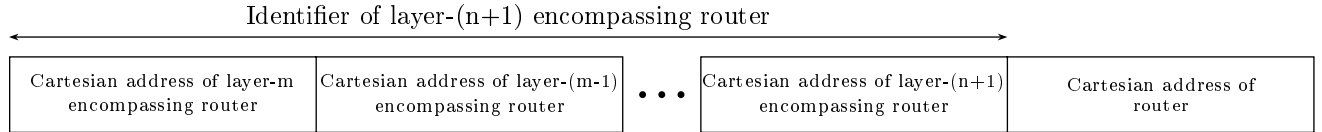


Figure 3: Hierarchical identifier structure for a layer- $n$  collector router

Every router in the hierarchy has a unique address, meaning that a router in a network at layer- $n$  can determine whether or not a packet is local to the network. A packet is said to be **local** to a network if the network encompasses the destination address of the packet. A router can determine this by comparing the most significant  $m-n$  Cartesian addresses of the packet’s destination address with the first  $m-n$  Cartesian addresses of its own identifier.

### 2.3 Packet Routing

A packet can enter a network at layer- $n$  through the bottom port of a collector router or the top port of the network’s IR. Packets received on the bottom port of a collector router are either local or non-local to the network. When a router finds a packet to be local, the packet is “tagged” as a local packet by setting a bit in the packet’s address called the **local** bit. When a packet is local to a network at layer- $n$ , the  $(m-n+1)^{\text{th}}$  Cartesian address of the packet’s destination address is used to route the packet in the network using Cartesian routing algorithm. For example, at layer- $m$ , the first Cartesian address is used for Cartesian routing, while at layer-1 the  $m^{\text{th}}$  address is used. When a packet is received by a router on its bottom port, the address is inspected: non-local packets are forwarded to the network’s IR in order to be delivered to the encompassing network.

Forwarding a packet to the IR requires each collector router and arterial to maintain an **Internetwork Router Direction Indicator** or IRDI that indicates which port leads to the IR. If the packet is determined to be non-local, it is forwarded in the direction specified by the IRDI. When IRDI indicates that IR is not accessible, the packet is dropped and a message is returned to the source notifying that the destination address is not reachable.

A collector router that receives a packet on its west or east port checks the local bit; if set, the Cartesian routing algorithm is employed to route the packet, otherwise the packet is forwarded to the opposite port. When a packet enters a network through the top port of the network’s IR, the packet is guaranteed to be local, since this has already been verified by the encompassing network. The IR sets the local bit of the

packet's address upon receiving it on its top port and then applies the Cartesian routing algorithm on the  $(m-n+1)^{\text{th}}$  Cartesian address of the packet's destination address.

## 2.4 Terrestrial Example

The Earth is nearly spherical, with an equatorial circumference of 40,075 km and a polar circumference of 39,940 km. The first step in developing the network is to project the Earth onto a two-dimensional surface, 40,075 km by 39,940 km. This is the *world* layer.

Since Cartesian routing is to take place at the world-layer, it is necessary to divide this layer into a grid of horizontal lines (the latitudes) and vertical lines (the longitudes). For simplicity, by using two bytes to represent the address of a layer (i.e., one byte for the latitude and the other for longitude), a layer can be divided into  $2^{16}$  grid elements (subsequently referred to as *cantons*). Each canton is associated with its own inter-regional router.

At the world layer, each canton measures approximately 156 km by 156 km. These cantons, referred to as *areas*, can also be subdivided. If two bytes are used to identify cantons within an area, each canton (a *region*) measures about 611 metres by 611 metres. Subdividing a region into its cantons, results in a canton about 2.4 metres by 2.4 metres (a *subregion*). (A further subdivision of a subregion would reduce the size of a canton to about 9.4 mm by 9.4 mm, this size of canton is not considered at this time.)

A four-layer model, with byte-pairs addressing cantons within each layer, allows a subregion to be identified on the Earth using only six bytes. Table 1 illustrates the different layers, their sizes, and possible examples.

Table 1: Four-layer Cartesian network for the Earth

Layer	Name	Size	Example
4	World	40,075 by 39,940 km	The entire planet
3	Area	156 by 156 km	Small country, state, or province
2	Region	611 by 611 metres	Village or several city blocks
1	Subregion	2.4 by 2.4 metres	Small room

## 3 Future Research

The following is a description of some of the issues relating to Cartesian routing presently under consideration by the authors:

**Optical Routers.** At present, any optical transmission must be converted into its electrical equivalent before any routing decisions can be made. By avoiding this conversion, packet transmission times would fall by several orders of magnitude.

As demonstrated by the prototype collector described in section 1.1.1, Cartesian routing decisions can be made in  $O(1)$  time by a simple comparison of a packet's address with the router's address. This is faster than existing routing technology without any change in hardware technology. Moving the design to FPGAs or ASICs could reduce the overall packet transmission time still further; however, even if coupled with an optical network, Cartesian routing would still be penalized as the internal media remains electrical.

Major advances have taken place over the past several years in the field of optical logic. The simplicity of both the collector and arterial router design, coupled with optical logic, would further reduce packet transmission times.

**Network Evolution.** With the exception of newly created subnetworks, it is unrealistic to expect existing ISPs and other Internet players to rewire their networks to conform to the Cartesian network topology. However, by using a multiple layer Cartesian network as the internetwork's backbone, layer-1 networks (consisting of existing technology) could be interconnected.

**Congestion Control.** The simplicity of the Cartesian routing algorithm means that packets follow fixed routes through the network until a link or router fails, at which point, a different path may be taken. Since minimal state information is exchanged between routers, routers have a limited view of the overall network

state; this can lead to congestion and degraded network service. The authors are considering techniques of sharing link state information with minimal impact upon the routing algorithm.

At a minimum, a router could detect congestion on a link and then forward packets on other links, leading to the destinations. For example, an arterial router could use a north-link in place of a northwest-link if the northwest-link was reaching congestion; the new path might be geographically longer than the original, but could prove faster. This approach can be extended by allowing arterial routers to exchange information regarding the state of a link several hops away.

**Network Interconnection.** The lowest layer in a four-layer terrestrial network maps into a subregion about 2.4 by 2.4 metres using only three byte-pairs (48 bits). If 128-bit IPv6 addresses are employed to identify the subregion, part of the remaining 80 bits could be used for addressing within the subregion [3]. For example, there are sufficient bits for subregion networks using Bluetooth or using EUI-64. The authors intend to add a Bluetooth gateway to the bottom port of a collector router to examine network interconnection issues.

**Interregional Routers.** In the present multiple-layer Cartesian network design, no consideration is given to the placement of inter-regional routers (IR) within a canton; in a ‘real’ implementation it will be necessary to perform some form of traffic analysis for the optimal placement of IRs.

An additional problem arises since there is only one IR per canton, meaning that the IR is a potential single point of failure. This limitation must be viewed from the canton and its encompassing layer: within the canton, the presense of multiple IRs would require changes to the way IRs are identified. Outside the canton, it would be necessary to adopt some form of aliasing to overcome the Cartesian positional requirements.

**Other Issues.** Three other research areas that the authors plan to investigate include the placement and number of collector routers between arterial routers, the development of a simulator for wide area networks, and an extension to Cartesian routing known as Quadtree routing.

## Acknowledgements

The authors would like to thank their colleagues Sandy Cook, Evan Hughes, Yanting Lin, Ryan Murphy, and Shehzad Pradhan for their contributions to the project and this paper.

## References

- [1] D. Comer. *Internetworking with TCP/IP*. Prentice-Hall, 1988.
- [2] D. Comer. *Computer Networks and Internets*. Prentice-Hall, 1997.
- [3] S. Deering and R. Hinden. *Internet Protocol, Version 6 (IPv6) Specification*. RFC 1883.
- [4] K. Estabrooks and P. Poirier. A Cartesian Router, 2000. Senior year computer engineering project.
- [5] L. Hughes. Teaching Data Communications to Computer Science Students. *SIGCSE Bulletin*, 21(1), February 1989.
- [6] L. Hughes. *Introduction to Data Communications: A Practical Approach*. Jones and Bartlett Publishers, Boston, 1996.
- [7] L. Hughes, O. Banyasad, and E. Hughes. Cartesian Routing. *Computer Networks*, 34(3):455–466, 2000.
- [8] L. Hughes, O. Banyasad, and E. Hughes. Wide Area Cartesian Routing. January 2001. Submitted to *Computer Networks*.
- [9] C. Huitema. *IPv6 - The New Internet Protocol*. Prentice Hall, second edition, 1997.
- [10] M. Reid. Cisco Systems. Personal communication, March 2000.
- [11] C. Semeria. Understanding IP Addressing: Everything You Ever Wanted to Know, April 1996. [www.3com.com/nscl50/302.html](http://www.3com.com/nscl50/302.html).