



CISCO SYSTEMS, INC. Petitioner,
V.
PORTSMOUTH NETWORK CORP., Patent Owner

Case No. IPR2024-00503
U.S. Patent No. 7,035,279

Theodore Foster
Haynes and Boone, LLP

May 5, 2025



'279 Patent – Claim 1

1. In a communication network that includes a plurality of nodes interconnected by links, the nodes including at least first and second nodes, a method for routing a data flow, comprising:

- allocating a respective number of quanta of one or more resources in the network to each of the nodes;

- receiving a request at the first node to use a portion of the resources so as to carry the data flow between the first and second nodes over one of a plurality of paths therebetween, each such path comprising a respective sequence of the links; and

- in response to the request:

 - if the resources already allocated to the first node on one of the paths are sufficient, directing the data flow from the first node to the second node over the one of the paths without requesting additional resources; and

 - if the resources already allocated to the first node are insufficient, performing the following steps:

 - requesting an increase in an allocation to the first node of the requested resources;

 - determining, for the links comprised by each of the paths, respective levels of use of the requested resources due to communications in progress over the network;

 - selecting which of the paths is to carry the data flow responsive to the determined levels of use of the requested resources on the links comprised in each of the paths; and

 - increasing the allocation of the resources to the first node on the selected path by a predetermined quantum.

Key Points

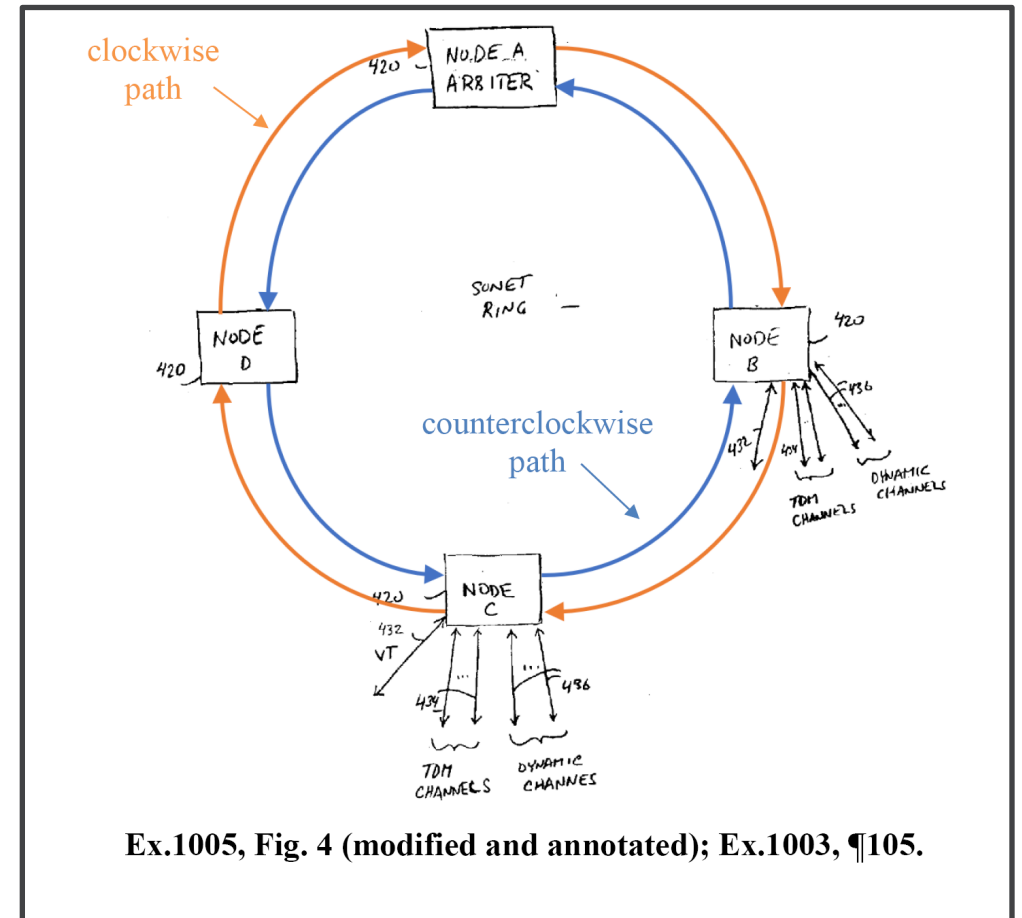
- Obvious to combine Kovvali and Kalman
- A best-effort data packet is a “*request to use... resources*”
- Kovvali’s 9-byte columns are “*predetermined quanta*”

Kovvali + Kalman: Obvious to combine

Petition

[1.6] determining, for the links comprised by each of the paths, respective levels of use of the requested resources due to communications in progress over the network;

First, as discussed in [1.2], Kovvali's teaching of a bidirectional ring network provides two paths for transmitting data between nodes, illustrated in modified Figure 4 below. There is a first path in a clockwise direction (blue) and a second path in a counterclockwise direction (orange). It would have been obvious to a POSITA that when the arbiter receives a request to increase the bandwidth allocated for communications from a first node to a second node, the arbiter would have two path options to choose from for potentially accommodating that bandwidth increase request. Ex.1003, ¶105.



Ex.1005, Fig. 4 (modified and annotated); Ex.1003, ¶105.

Kovvali + Kalman: Obvious to combine

Petition

Second, Kalman discloses a technique for selecting a “preferred direction for traffic” between two nodes, “based on the direction with the lowest cost.”

Ex.1006, 7:50-53; Ex.1003, ¶106. Like Kovvali, Kalman discloses a network that “includes two rings, wherein a first ring transmits data in a ‘clockwise’ direction (or direction 0), and the other ring transmits data in a ‘counterclockwise’ direction (or direction 1 opposite of direction 0).” Ex.1006, 2:61-63. “[D]ata between nodes flows on the ring that provides the lowest-cost path to the destination node,” Ex.1006, 3:2-3, “rather than in both directions,” Ex.1006, 4:24-26, Ex.1003, ¶107.

have been within the POSITA’s skillset. A POSITA would have appreciated that by having Kovvali’s arbiter direct data transmission toward the least-occupied or less congested portion of the ring network (e.g., to the clockwise or counterclockwise portion that has more unallocated bandwidth), more efficient utilization of the available network capacity would be achieved. Alternatively, a POSITA would have appreciated that having Kovvali’s arbiter direct data transmission toward the most-occupied or more congested portion of the ring network (e.g., to the clockwise or counterclockwise portion that has more allocated bandwidth) would leave a larger contiguous block of unallocated space on the other portion of the network (e.g., in the opposite direction), thus making it possible to accommodate a larger bandwidth request that may arrive in the future.

Kovvali + Kalman: Obvious to combine

Patent Owner Response

1006, 3:9. Thus, the communication and network structure in Kovvali is completely different than that in Kalman, and is designed to achieve different outcomes. Ex.

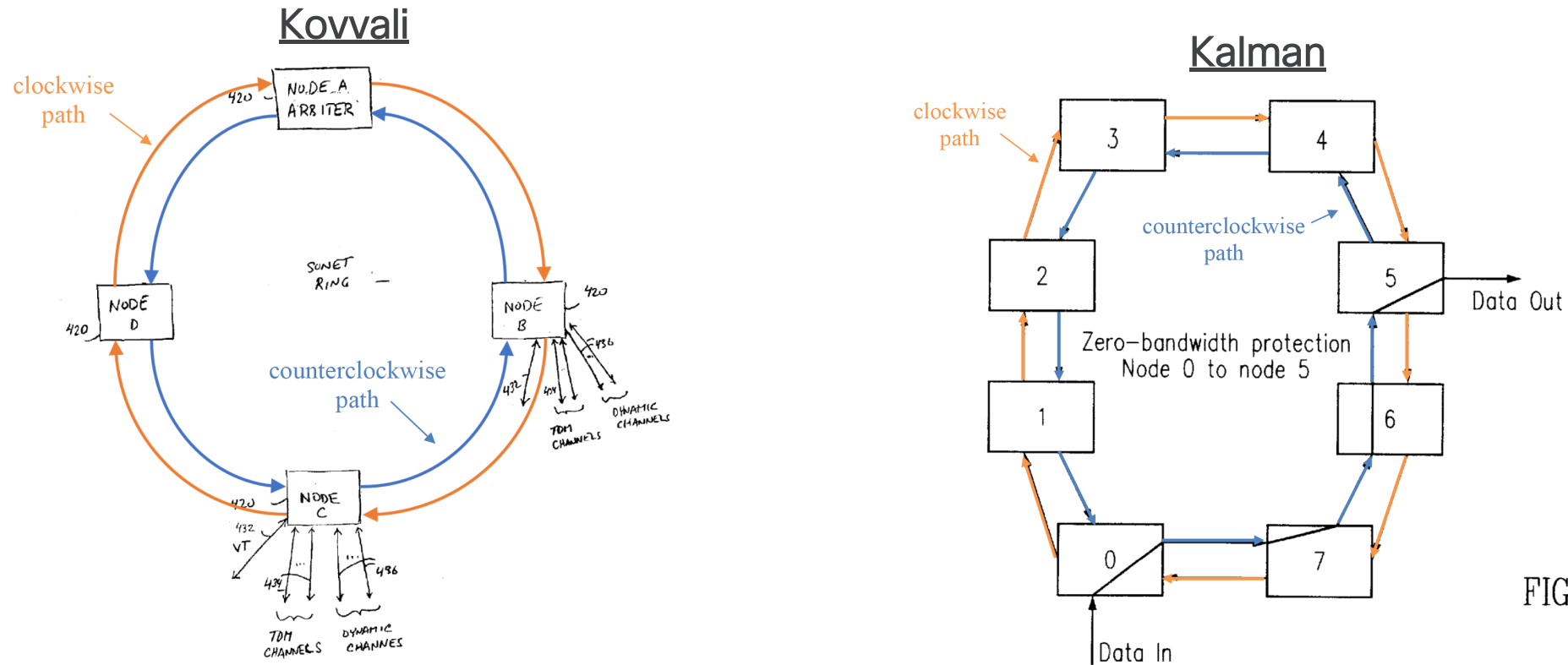
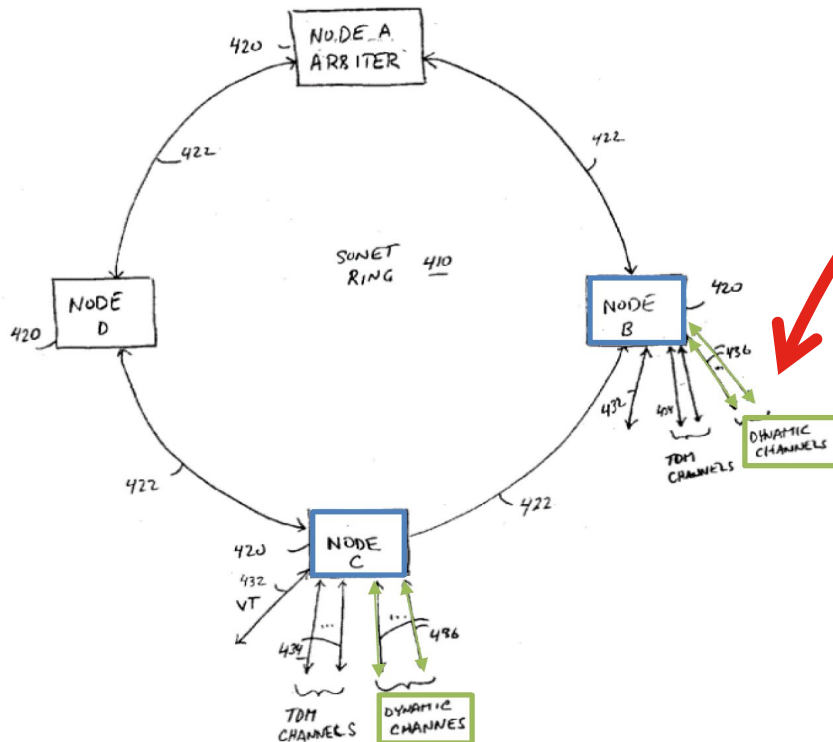


FIG. 3

“receiving a request... to use... resources”

Petition



Ex.1005, Fig. 4 (annotated); Ex.1003, ¶80.

Accordingly, a POSITA would have understood that the receipt of data from local dynamic channels by a node (e.g. Node B or Node C) for transmission to another node, renders obvious “receiving a request at the first node to use a portion of the resources” because it is a request for the node to process (e.g. queue, schedule, police) and multiplex the data onto the portion of network resources allocated by the arbiter (as discussed in [1.1]) to the node for the respective dynamic channels in the dynamic frame. Ex.1003, ¶83.

allocated.” Ex.1005, 9:48-66; Ex.1003, ¶70. The dynamic channels in the elastic section are “flexible bandwidth channels” where bandwidth “allocations are in units of columns of the dynamic frame.” Ex.1005, 4:43-46, 12:6-7; Ex.1003, ¶¶71-72.

Kovvali allocates bandwidth in *predetermined quanta*

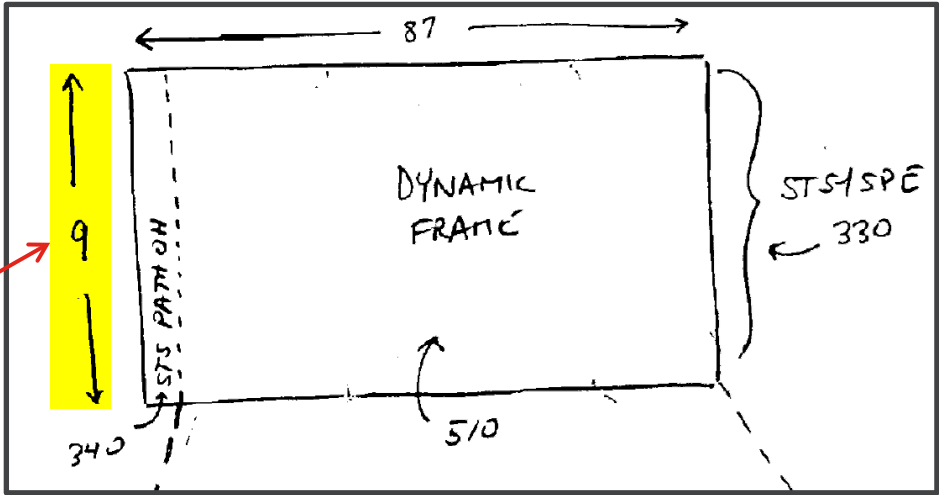
Petition

Furthermore, it would have been obvious to a POSITA that allocating bandwidth in “units of columns” is allocating bandwidth in “*quanta*” because each column is the smallest discrete unit of bandwidth that can be allocated to a dynamic channel. Ex.1003, ¶74. Referring to Figure 5A discussed above, each column of bandwidth corresponds to 9 rows of 8-bit bytes; therefore, each column corresponds to 72 bits. Ex.1005, 2:56-58; Ex.1003, ¶74. Two columns of

9 rows of 8-bit bytes.

Kovvali

REQ	Requested Bandwidth
000	No bandwidth required
001	Deallocate 1 column
010	Deallocate 3 columns
011	Deallocate 6 columns
100	Allocate 1 additional column
101	Allocate 3 additional columns
110	Allocate 6 additional columns
111	Allocate 10 additional columns





The bridge to possible