BRIEF SUMMARY

[0004] One aspect of the present disclosure determines bandwidth allocation of inter-block traffic in a data center network across a set of end-to-end paths based on a variety of information. It employs a number of optimization objectives and a heuristic water-filling strategy to avoid producing unnecessary paths and to avoid determining paths that would be unavailable when actually needed.

[0004a] According to a first aspect, there is provided a method of traffic engineering in a computer network having a plurality of fabric blocks in communication with one another, the method comprising:

grouping, by one or more processors, a plurality of middle blocks into a plurality of logical clusters, each of the plurality of logical clusters forming one of the plurality of fabric blocks and including at least two middle blocks;

identifying at least one destination fabric block from among the plurality of fabric blocks;

evaluating, in round-robin order, each destination fabric block of the plurality of fabric blocks;

choosing in round-robin order, by the one or more processors, a single destination middle block MB_d in a selected destination fabric block FB_d;

selecting in round-robin order, by the one or more processors, a source fabric block FB_s;

identifying, by the one or more processors, a subset M of source middle blocks from among all middle blocks in the source fabric block FB_s, the subset of middle blocks being identified as having a shortest available path length to the destination middle block MB_d; and

selecting for use in a network path, by the one or more processors, one of the source middle blocks MB_s in the subset M that maximizes the following condition:

min(demand(MB_s, FB_d), spf_capacity(MB_s, MB_d)),

in which spf_capacity(MB_s, MB_d) represents a shortest path capacity between the source middle block MB_s and the destination middle block MB_d, and demand(MB_s, FB_d) represents a demand between the source middle block MB_s and the destination fabric block FB_d.

[0004b] According to a second aspect, there is provided a system comprising one or more processors configured to:

group a plurality of middle blocks into a plurality of logical clusters, each of the plurality of logical clusters forming a fabric block including at least two middle blocks;

identify at least one destination fabric block from among the plurality of fabric blocks;

evaluate, in round-robin order, each destination fabric block of the plurality of fabric blocks;

choose, in round-robin order, a single destination middle block MB_d in a selected destination fabric block FB_d;

select, in round-robin order, a source fabric block FB_s;

identify a subset M of source middle blocks from among all middle blocks in the source fabric block FB_s, the subset of middle blocks being identified as having a shortest available path length to the destination middle block MB_d; and

select, for use in a network path, one of the source middle blocks MB_s in the subset M that maximizes the following condition:

min(demand(MB_s, FB_d), spf_capacity(MB_s, MB_d)),

in which spf_capacity(MB_s, MB_d) represents a shortest path capacity between the source middle block MB_s and the destination middle block MB_d, and demand(MB_s, FB_d) represents a demand between the source middle block MB_s and the destination fabric block FB_d.

[0004c] According to a third aspect, there is provided a non-transitory storage medium storing instructions

that, when executed by one or more processors, cause the one or more processors to perform a method of traffic engineering in a computer network having a plurality of fabric blocks in communication with one another, and one or more middle blocks in each fabric block, the method comprising:

grouping a plurality of middle blocks into a plurality of logical clusters, each of the plurality of logical clusters forming one of the plurality of fabric blocks and including at least two middle blocks;

identifying at least one destination fabric block from among the plurality of fabric blocks;

evaluating, in round-robin order, each destination fabric block of the plurality of fabric blocks;

choosing in round-robin order, by the one or more processors, a single destination middle block MB_d in a selected destination fabric block FB_d;

selecting in round-robin order, by the one or more processors, a source fabric block FB_s;

identifying, by the one or more processors, a subset M of source middle blocks from among all middle blocks in the source fabric block FB_s, the subset of middle blocks being identified as having a shortest available path length to the destination middle block MB_d; and

selecting for use in a network path, by the one or more processors, one of the source middle blocks MB_s in the subset M that maximizes the following condition:

$$\min(\text{demand}(MB_s, FB_d), \text{spf_capacity}(MB_s, MB_d)),$$

in which $\text{spf_capacity}(MB_s, MB_d)$ represents a shortest path capacity between the source middle block MB_s and the destination middle block MB_d, and $\text{demand}(MB_s, FB_d)$ represents a demand between the source middle block MB_s and the destination fabric block FB_d.

[0005] ~~According to one aspect of the disclosure,~~ Herein disclosed is a method of traffic

engineering in a computer network ~~is provided~~. ~~the~~ The computer network has a plurality of fabric blocks in communication with one another, and one or more middle blocks in each fabric block. The method comprises identifying, by one or more processors, a source fabric block FB_s from among the plurality of fabric blocks; evaluating, in round-robin order, each destination fabric block of the plurality of fabric blocks; choosing in round-robin order, by the one or more processors, a single destination middle block MB_d in a selected destination fabric block FB_d; selecting in round-robin order, by the one or more processors, a source fabric block FB_s; identifying, by the one or more processors, a subset M of source middle blocks from among all middle blocks in the source fabric block FB_s, the subset of middle blocks being identified as having a shortest available path length to the destination middle block MB_d; and selecting for use in a network path, by the one or more processors, one of the source middle blocks MB_s in the subset M that maximizes the following condition:

$$\min(\text{demand}(MB_s, FB_d), \text{spf_capacity}(MB_s, MB_d)),$$

in which $\text{spf_capacity}(MB_s, MB_d)$ represents a shortest path capacity between the source middle block MB_s and the destination middle block MB_d, and $\text{demand}(MB_s, FB_d)$ represents a demand between the source middle block MB_s and the destination fabric block FB_d.

CLAIMS

1. A method of traffic engineering in a computer network having a plurality of fabric blocks in communication with one another~~, and one or more middle blocks in each fabric block~~, the method comprising:

grouping, by one or more processors, a plurality of middle blocks into a plurality of logical clusters, each of the plurality of logical clusters forming one of the plurality of fabric blocks and including at least two middle blocks;

identifying~~, by one or more processors, a source~~ at least one destination fabric block~~ FB_s~~ from among the plurality of fabric blocks;

evaluating, in round-robin order, each destination fabric block of the plurality of fabric blocks;

choosing in round-robin order, by the one or more processors, a single destination middle block MB_d in a selected destination fabric block FB_d;

selecting in round-robin order, by the one or more processors, a source fabric block FB_s;

identifying, by the one or more processors, a subset M of source middle blocks from among all middle blocks in the source fabric block FB_s, the subset of middle blocks being identified as having a shortest available path length to the destination middle block MB_d; and

selecting for use in a network path, by the one or more processors, one of the source middle blocks MB_s in the subset M that maximizes the following condition:

$$\min(\text{demand}(MB_s, FB_d), \text{spf_capacity}(MB_s, MB_d)),$$

in which $\text{spf_capacity}(MB_s, MB_d)$ represents a shortest path capacity between the source middle block MB_s and the destination middle block MB_d, and $\text{demand}(MB_s, FB_d)$ represents a demand between the source middle block MB_s and the

destination fabric block FB_d.

2. The method of claim 1, further comprising generating a route computation graph that facilitates bandwidth allocation through iterative selection of source and destination middle blocks maximizing the condition.

group a plurality of middle blocks into a plurality of logical clusters, each of the plurality of logical clusters forming a fabric block including at least two middle blocks;

identify ~~a source~~at least one destination fabric block ~~FB_s~~ from among ~~a~~ the plurality of fabric blocks ~~in communication with one another in a computer network~~;

evaluate, in round-robin order, each destination fabric block of the plurality of fabric blocks;

choose, in round-robin order, a single destination middle block MB_d in a selected destination fabric block FB_d;

select, in round-robin order, a source fabric block FB_s;

identify a subset M of source middle blocks from among all middle blocks in the source fabric block FB_s, the subset of middle blocks being identified as having a shortest available path length to the destination middle block MB_d; and

select, for use in a network path, one of the source middle blocks MB_s in the subset M that maximizes the following condition:

$$\min(\text{demand}(MB_s, FB_d), \text{spf_capacity}(MB_s, MB_d)),$$

in which $\text{spf_capacity}(MB_s, MB_d)$ represents a shortest path capacity between the source middle block MB_s and the destination middle block MB_d, and $\text{demand}(MB_s, FB_d)$ represents a demand between the source middle block MB_s and the destination fabric block FB_d.

11. The system of claim 10, wherein the one or more processors are further configured to generate a route computation graph that facilitates bandwidth allocation through iterative selection of source and destination middle blocks maximizing the condition.

12. The system of claim 10, wherein the one or more

processors are further configured to, for each source fabric block, split demand to a destination fabric block equally among each non-faulty middle block in the destination fabric block.

engineering in a computer network having a plurality of fabric blocks in communication with one another, and one or more middle blocks in each fabric block, the method comprising:

grouping a plurality of middle blocks into a plurality of logical clusters, each of the plurality of logical clusters forming one of the plurality of fabric blocks and including at least two middle blocks;

identifying~~, by one or more processors, a source~~ at least one destination fabric block~~ FB_s~~ from among the plurality of fabric blocks;

evaluating, in round-robin order, each destination fabric block of the plurality of fabric blocks;

choosing in round-robin order, by the one or more processors, a single destination middle block MB_d in a selected destination fabric block FB_d;

selecting in round-robin order, by the one or more processors, a source fabric block FB_s;

identifying, by the one or more processors, a subset M of source middle blocks from among all middle blocks in the source fabric block FB_s, the subset of middle blocks being identified as having a shortest available path length to the destination middle block MB_d; and

selecting for use in a network path, by the one or more processors, one of the source middle blocks MB_s in the subset M that maximizes the following condition:

$$\min(\text{demand}(MB_s, FB_d), \text{spf_capacity}(MB_s, MB_d)),$$

in which $\text{spf_capacity}(MB_s, MB_d)$ represents a shortest path capacity between the source middle block MB_s and the destination middle block MB_d, and $\text{demand}(MB_s, FB_d)$ represents a demand between the source middle block MB_s and the destination fabric block FB_d.

20. The non-transitory storage medium of claim 19, wherein the method further comprises generating a route computation graph that facilitates bandwidth allocation through iterative selection of source and destination middle blocks maximizing the condition.