Claims

1. A classification-based virtual network embedding method, wherein the method comprises:

step 101) when receiving a virtual network request, classification virtual nodes related to a virtual network request according to the classification standard, and dividing the virtual nodes belonging to different classifications to different virtual network subrequests;

wherein the classification standard is namely to classify the virtual nodes into a substrate/physical network of an area nearest thereof according to the demand information of the virtual nodes on geographical positions; and all the virtual nodes in the same virtual network subrequest belong to the same classification, and the classification is namely a classification to which any virtual node in the virtual network subrequest belongs;

step 102) sending each virtual network subrequest to a certain area in the substrate/physical network to which the virtual network subrequest belongs to; and

step 103) mapping the virtual network subrequests onto the bottom physical layer network in the area according to a load balancing demand of each node in the virtual network subrequest and the load balancing capacity of each node in the area; wherein,

the load balancing demand represents the load demand of each virtual node in case of balanced loads; and the load demand is defined as a product of resources of the virtual node request and resources of all link requests connected to the virtual node; and

the load balancing capacity represents the load capacity of each physical node in the area in case of balanced load; and the load capacity is defined as a product of available resources of the physical node and available resources of all links connected to the physical node.

2. The classification-based virtual network embedding method according to claim 1 wherein step 103) further comprises:

step 103-1) detecting a virtual node n^v which is unmapped currently and has maximum load balancing demand in the virtual network subrequest;

step 103-2) seeking k candidate bottom physical nodes of the virtual node n^v in the area, wherein the k candidate nodes are the first k bottom physical nodes with maximum load balancing capacity which are unmapped in the area currently and can satisfy the demand of the virtual node n^v; and

step 103-3) for each candidate bottom physical node, calculating the sum of the path length between the candidate bottom physical node and the bottom physical nodes corresponding to each mapped virtual node, mapping the virtual node n^v on the candidate node having shortest path length, and mapping the virtual links between the mapped virtual nodes.

3. The classification-based virtual network embedding method according to claim 1 or 2, wherein the load balancing capacity is obtained as follows:

step 201) respectively performing normalization processing on resources of the physical nodes and physical links of each area inclusive in the substrate/physical network;

wherein, the normalization processing of the physical nodes is related to the storage capacity of the nodes, magnetic disk I/O and quantity of CPU, and memory size; and the normalization processing of the physical links is related to the band width, delay and delay jitter of the links;

step 202) calculating the load capacity of the nodes in each area in the substrate/physical network;

wherein, the load capacity is defined as a product of available resources of the physical node and available resources of all links connected to the physical node.

step 203) calculating load balancing matrix of each area in the substrate/physical network;

wherein, the elements of the balanced load matrix are: a weighted value mutually contributed by the load balancing capacities between the physical nodes in a certain area, and the weighted value mutually contributed by the load balancing capacities is particularly as follows: when the physical node n^s is just the physical node , i.e.,

$n^s = m^s$, or the physical node n^s and the physical node m^s are connected by a physical link, the physical node n^s contributes to the load balancing capacity of the physical node m^s, and the weighted value is a nonzero positive number at the moment; otherwise, the weight contributed by the physical node n^s to the load balancing capacity of the physical node m^s is 0; wherein, n^s represents the n^s line of the matrix, m^s represents the m^s row of the matrix, both the line number and the row number of the matrix are the total number of the physical nodes, and a corner mark s represents the substrate/physical network; and

step 204) iteratively calculating the load balancing capacity of each physical node in the area according to the load balancing matrix, i.e., the load balancing capacity of the physical node n^s is the sum of the load balancing capacities contributed by all the physical nodes in the area.

4. The classification-based virtual network embedding method according to claim 1 or 4, wherein the load balancing demand is obtained by employing the following method:

step 301 respectively performing normalization processing on the virtual node and virtual link demands in each virtual network subrequest;

wherein, the normalization processing of the virtual nodes is related to the demands on the storage capacity of the nodes, quantity of magnetic disk I/O and CPU, and memory size; and the normalization processing of the virtual links is related to the demands on the band width, delay and delay jitter of the links;

step 302) calculating the load demand of each virtual node in each virtual network subrequest;

wherein, the load demand is defined as a product of demands of the virtual node and demands of all links connected to the virtual node;

step 303) calculating the load balancing matrix of each virtual network subrequest;

wherein, the elements of the load balancing matrixes are: a weighted value mutually

affected by the load balancing demands between the virtual nodes in the virtual network subrequest, and the weighted value is particularly as follows: when the virtual node n^v is just the virtual node m^v, i.e., $n^v = m^v$, or the virtual node n^v and the virtual node m^v are connected by a virtual link, the weighted value is defined to be a nonzero positive number; otherwise, the weight affected by n^v to the load balancing demand of m^v is 0;

wherein, n^v represents the n^v line of the matrix, m^v represents the m^v row of the matrix, both the line number and the row number of the matrix are the total number of the virtual nodes, and a corner mark v represents the virtual network request; and

step 304) iteratively calculating the load balancing demand of each virtual node in the area according to the load balancing matrix, i.e., the load balancing demand of the virtual node n^v being the sum of the load balancing demands affected by all the virtual nodes in the virtual network subrequest.

5. A classification-based virtual network embedding system, wherein the system comprises:

a global management module configured to, when receiving a virtual network request, classify virtual nodes related to the virtual network request according to a classification standard, and divide the virtual nodes belonging to different classifications to different virtual network subrequests, wherein the classification standard means that one virtual node classifies the virtual nodes into a substrate/physical network of an area nearest thereof; and send each virtual network subrequest to a certain area in the substrate/physical network to which the virtual network subrequest belongs to; and

a local management module configured to map the virtual network subrequests onto the bottom physical layer network in the area according to a load balancing demand of each node in the virtual network subrequest and the load balancing capacity of each node in the area;

wherein, the load balancing demand represents the load demand of each virtual

node in case of balanced loads; and the load demand is defined as a product of resources of the virtual node request and resources of all link requests connected to the virtual node;

the load balancing capacity represents the load capacity of each physical node in the area in case of balanced load; and the load capacity is defined as a product of available resources of the physical node and available resources of all links connected to the physical node.

6. The classification-based virtual network embedding system according to claim 5, wherein the local management module further comprises:

a load balancing capacity or load balancing demand acquisition submodule configured to acquire the load balancing capacity value of each node in each area of the substrate/physical network, and acquire the load balancing demand value of each virtual node of each virtual network subrequest;

a detection submodule configured to detect a virtual node n^v which is unmapped currently and has maximum load balancing demand in the virtual network subrequest;

a candidate bottom physical node selection submodule configured to seek k candidate bottom physical nodes of the virtual node n^v in the area, wherein the k candidate nodes are the first k bottom physical nodes with maximum load balancing capacity which are unmapped in the area currently and can satisfy the demand of the virtual node n^v; and

a node and link mapping submodule configured to, for each candidate bottom physical node, calculate the sum of the path length between the candidate bottom physical node and the bottom physical nodes corresponding to each mapped virtual node, map the virtual node n^v on the candidate node having shortest path length, and map the virtual links between the mapped virtual nodes.

7. The classification-based virtual network embedding system according to claim 6, wherein the load balancing capacity acquisition submodule further comprises:

a first normalization processing module configured to respectively perform

normalization processing on resources of the physical nodes and physical links of each area inclusive in the substrate/physical network;

wherein, the normalization processing of the physical nodes is related to the storage capacity of the nodes, magnetic disk I/O and quantity of CPU, and memory size; and the normalization processing of the physical links is related to the band width, delay and delay jitter of the links;

a first node load capacity acquisition module configured to calculate the load capacity of each node in the substrate/physical network;

wherein, the load capacity is defined as a product of available resources of the physical node and available resources of all links connected to the physical node;

a load balancing matrix acquisition module configured to calculate the load balancing matrix of each node in the substrate/physical network;

wherein, the load balancing matrix defines a weighted value mutually contributed by the load balancing capacities between the physical nodes in the area of the substrate/physical network, when and only when the physical node n^s is just the physical node , i.e., $n^s = m^s$, or the physical node n^s and the physical node m^s are connected by a physical link, the weighted value is a nonzero positive number at the moment; otherwise, the weight contributed by the physical node n^s to the load balancing capacity of the physical node m^s is 0;

a load balancing capacity calculation module configured to iteratively calculate the load balancing capacity of each physical node in the area according to the load balancing matrix, i.e., the load balancing capacity of the physical node n^s is the sum of the load balancing capacities contributed by all the physical nodes in the area.

8. The classification-based virtual network embedding system according to claim 6, wherein the load balancing demand acquisition submodule further comprises:

a second normalization processing module configured to respectively perform normalization processing on the virtual node and virtual link demands in each virtual network subrequest;

wherein, the normalization processing of the virtual nodes is related to the demands on the storage capacity of the nodes, quantity of magnetic disk I/O and CPU, and memory size; and the normalization processing of the virtual links is related to the demands on the band width, delay and delay jitter of the links;

a second load demand calculation module configured to calculate the load demand of each virtual node in each virtual network subrequest;

wherein, the load demand is defined as a product of demands of the virtual node and demands of all links connected to the virtual node;

a second load balancing matrix calculation module configured to calculate the load balancing matrix of each virtual network subrequest;

wherein, the load balancing matrix defines a weighted value mutually affected by the load balancing demands between the virtual nodes in the virtual network subrequest, when and only when the virtual node n^v is just the virtual node m^v, i.e., $n^v = m^v$, or the virtual node n^v and the virtual node m^v are connected by a virtual link, the weighted value is a nonzero positive number at the moment; otherwise, the weight affected by the virtual node n^v to the load balancing demand of the virtual node m^v is 0; and

a load balancing demand acquisition module configured to iteratively calculate the load balancing demand of each virtual node in the area according to the load balancing matrix, i.e., the load balancing demand of the virtual node n^v being the sum of the load balancing demands affected by all the virtual nodes in the virtual network subrequest.

Claims

1. A classification-based virtual network embedding method, wherein the method comprises:

step 101) when receiving a virtual network request, classification virtual nodes related to a virtual network request according to the classification standard, and dividing the virtual nodes belonging to different classifications to different virtual network subrequests;

wherein the classification standard is namely to classify the virtual nodes into a substrate/physical network of an area nearest thereof according to the demand information of the virtual nodes on geographical positions; and all the virtual nodes in the same virtual network subrequest belong to the same classification, and the classification is namely a classification to which any virtual node in the virtual network subrequest belongs;

step 102) sending each virtual network subrequest to a certain area in the substrate/physical network to which the virtual network subrequest belongs to; and

step 103) mapping the virtual network subrequests onto the bottom physical layer network in the area according to a load balancing demand of each node in the virtual network subrequest and the load balancing capacity of each node in the area; wherein,

the load balancing demand represents the load demand of each virtual node in case of balanced loads; and the load demand is defined as a product of resources of the virtual node request and resources of all link requests connected to the virtual node; and

the load balancing capacity represents the load capacity of each physical node in the area in case of balanced load; and the load capacity is defined as a product of available resources of the physical node and available resources of all links connected to the physical node；

the load balancing demand is obtained by employing the following method:

step 301 respectively performing normalization processing on the virtual node and virtual link demands in each virtual network subrequest;

wherein, the normalization processing of the virtual nodes is related to the

demands on the storage capacity of the nodes, quantity of magnetic disk I/O and CPU, and memory size; and the normalization processing of the virtual links is related to the demands on the band width, delay and delay jitter of the links;

step 302) calculating the load demand of each virtual node in each virtual network subrequest;

wherein, the load demand is defined as a product of demands of the virtual node and demands of all links connected to the virtual node;

step 303) calculating the load balancing matrix of each virtual network subrequest;

wherein, the elements of the load balancing matrixes are: a weighted value mutually affected by the load balancing demands between the virtual nodes in the virtual network subrequest, and the weighted value is particularly as follows: when the virtual node n^v is just the virtual node m^v, i.e., $n^v = m^v$, or the virtual node n^v and the virtual node m^v are connected by a virtual link, the weighted value is defined to be a nonzero positive number; otherwise, the weight affected by n^v to the load balancing demand of m^v is 0;

wherein, n^v represents the n^v line of the matrix, m^v represents the m^v row of the matrix, both the line number and the row number of the matrix are the total number of the virtual nodes, and a corner mark v represents the virtual network request; and

step 304) iteratively calculating the load balancing demand of each virtual node in the area according to the load balancing matrix, i.e., the load balancing demand of the virtual node n^v being the sum of the load balancing demands affected by all the virtual nodes in the virtual network subrequest.

2. The classification-based virtual network embedding method according to claim 1 wherein step 103) further comprises:

step 103-1) detecting a virtual node n^v which is unmapped currently and has

maximum load balancing demand in the virtual network subrequest;

step 103-2) seeking k candidate bottom physical nodes of the virtual node n^v in the area, wherein the k candidate nodes are the first k bottom physical nodes with maximum load balancing capacity which are unmapped in the area currently and can satisfy the demand of the virtual node n^v; and

step 103-3) for each candidate bottom physical node, calculating the sum of the path length between the candidate bottom physical node and the bottom physical nodes corresponding to each mapped virtual node, mapping the virtual node n^v on the candidate node having shortest path length, and mapping the virtual links between the mapped virtual nodes.

3. The classification-based virtual network embedding method according to claim 1 or 2, wherein the load balancing capacity is obtained as follows:

step 201) respectively performing normalization processing on resources of the physical nodes and physical links of each area inclusive in the substrate/physical network;

wherein, the normalization processing of the physical nodes is related to the storage capacity of the nodes, magnetic disk I/O and quantity of CPU, and memory size; and the normalization processing of the physical links is related to the band width, delay and delay jitter of the links;

step 202) calculating the load capacity of the nodes in each area in the substrate/physical network;

wherein, the load capacity is defined as a product of available resources of the physical node and available resources of all links connected to the physical node.

step 203) calculating load balancing matrix of each area in the substrate/physical network;

wherein, the elements of the balanced load matrix are: a weighted value mutually contributed by the load balancing capacities between the physical nodes in a certain area, and the weighted value mutually contributed by the load balancing capacities is

particularly as follows: when the physical node n^s is just the physical node , i.e., $n^s = m^s$, or the physical node n^s and the physical node m^s are connected by a physical link, the physical node n^s contributes to the load balancing capacity of the physical node m^s, and the weighted value is a nonzero positive number at the moment; otherwise, the weight contributed by the physical node n^s to the load balancing capacity of the physical node m^s is 0; wherein, n^s represents the n^s line of the matrix, m^s represents the m^s row of the matrix, both the line number and the row number of the matrix are the total number of the physical nodes, and a corner mark s represents the substrate/physical network; and

step 204) iteratively calculating the load balancing capacity of each physical node in the area according to the load balancing matrix, i.e., the load balancing capacity of the physical node n^s is the sum of the load balancing capacities contributed by all the physical nodes in the area.

4. A classification-based virtual network embedding system, wherein the system comprises:

a global management module configured to, when receiving a virtual network request, classify virtual nodes related to the virtual network request according to a classification standard, and divide the virtual nodes belonging to different classifications to different virtual network subrequests, wherein the classification standard means that one virtual node classifies the virtual nodes into a substrate/physical network of an area nearest thereof; and send each virtual network subrequest to a certain area in the substrate/physical network to which the virtual network subrequest belongs to; wherein when classification analysis is performed on virtual nodes in a virtual network request, the virtual nodes are classified into a substrate/physical network of an area nearest thereof according to the demand information of the virtual nodes on geographical position; and

a local management module configured to map the virtual network subrequests

onto the bottom physical layer network in the area according to a load balancing demand of each node in the virtual network subrequest and the load balancing capacity of each node in the area;

wherein, the load balancing demand represents the load demand of each virtual node in case of balanced loads; and the load demand is defined as a product of resources of the virtual node request and resources of all link requests connected to the virtual node;

the load balancing capacity represents the load capacity of each physical node in the area in case of balanced load; and the load capacity is defined as a product of available resources of the physical node and available resources of all links connected to the physical node; the local management module further comprises:

a load balancing capacity or load balancing demand acquisition submodule configured to acquire the load balancing capacity value of each node in each area of the substrate/physical network, and acquire the load balancing demand value of each virtual node of each virtual network subrequest;

a detection submodule configured to detect a virtual node n^v which is unmapped currently and has maximum load balancing demand in the virtual network subrequest;

a candidate bottom physical node selection submodule configured to seek k candidate bottom physical nodes of the virtual node n^v in the area, wherein the k candidate nodes are the first k bottom physical nodes with maximum load balancing capacity which are unmapped in the area currently and can satisfy the demand of the virtual node n^v; and

a node and link mapping submodule configured to, for each candidate bottom physical node, calculate the sum of the path length between the candidate bottom physical node and the bottom physical nodes corresponding to each mapped virtual node, map the virtual node n^v on the candidate node having shortest path length, and map the virtual links between the mapped virtual nodes.

5. The classification-based virtual network embedding system according to claim 4, wherein the load balancing capacity acquisition submodule further comprises:

a first normalization processing module configured to respectively perform normalization processing on resources of the physical nodes and physical links of each area inclusive in the substrate/physical network;

wherein, the normalization processing of the physical nodes is related to the storage capacity of the nodes, magnetic disk I/O and quantity of CPU, and memory size; and the normalization processing of the physical links is related to the band width, delay and delay jitter of the links;

a first node load capacity acquisition module configured to calculate the load capacity of each node in the substrate/physical network;

wherein, the load capacity is defined as a product of available resources of the physical node and available resources of all links connected to the physical node;

a load balancing matrix acquisition module configured to calculate the load balancing matrix of each node in the substrate/physical network;

wherein, the load balancing matrix defines a weighted value mutually contributed by the load balancing capacities between the physical nodes in the area of the substrate/physical network, when and only when the physical node n^s is just the physical node , i.e., $n^s = m^s$, or the physical node n^s and the physical node m^s are connected by a physical link, the weighted value is a nonzero positive number at the moment; otherwise, the weight contributed by the physical node n^s to the load balancing capacity of the physical node m^s is 0;

a load balancing capacity calculation module configured to iteratively calculate the load balancing capacity of each physical node in the area according to the load balancing matrix, i.e., the load balancing capacity of the physical node n^s is the sum of the load balancing capacities contributed by all the physical nodes in the area.

6. The classification-based virtual network embedding system according to claim 4, wherein the load balancing demand acquisition submodule further comprises:

a second normalization processing module configured to respectively perform normalization processing on the virtual node and virtual link demands in each virtual network subrequest;

wherein, the normalization processing of the virtual nodes is related to the demands on the storage capacity of the nodes, quantity of magnetic disk I/O and CPU, and memory size; and the normalization processing of the virtual links is related to the demands on the band width, delay and delay jitter of the links;

a second load demand calculation module configured to calculate the load demand of each virtual node in each virtual network subrequest;

wherein, the load demand is defined as a product of demands of the virtual node and demands of all links connected to the virtual node;

a second load balancing matrix calculation module configured to calculate the load balancing matrix of each virtual network subrequest;

wherein, the load balancing matrix defines a weighted value mutually affected by the load balancing demands between the virtual nodes in the virtual network subrequest, when and only when the virtual node n^v is just the virtual node m^v, i.e., $n^v = m^v$, or the virtual node n^v and the virtual node m^v are connected by a virtual link, the weighted value is a nonzero positive number at the moment; otherwise, the weight affected by the virtual node n^v to the load balancing demand of the virtual node m^v is 0; and

a load balancing demand acquisition module configured to iteratively calculate the load balancing demand of each virtual node in the area according to the load balancing matrix, i.e., the load balancing demand of the virtual node n^v being the sum of the load balancing demands affected by all the virtual nodes in the virtual network subrequest.