CLAIMS

What is claimed is:

1. A method, comprising:

associating a first contention window having a first size with a first application operating on a node in a network;

associating a second contention window having a second size with a second application operating on the node, wherein the first size is different from the second size, and wherein the first and second contention windows are based on priorities of the first and second applications;

transmitting a packet from the first application on the node in the network, the transmission following elapsing of the first contention window;

waiting a timeout period without receipt of an acknowledgement of the packet;

obtaining a quality metric of a link between the node and a destination node of the transmitted packet;

checking a traffic density of a portion of the network including the link;

reducing a short retry count maximum and a long retry count maximum if the quality metric is below a first threshold and the traffic density exceeds a second threshold;

informing an upper layer of a transmission protocol of the quality metric;

resizing the first contention window from the first size, the resizing based on input including the quality metric of the link and the traffic density, wherein resizing the first contention window comprises:

distinguishing between transmission failures due to traffic collisions and transmission failures due to link quality issues;

resizing the first contention window if transmission failures are due to traffic collisions; and

retransmitting the packet after the resized first contention window elapses.

2. The method of claim 1, wherein obtaining the quality metric of the link comprises:

operating an expected transmission time algorithm on the node; and

calculating the quality metric based on the expected transmission time algorithm.

3. The method of claim 1, wherein checking the traffic density comprises:

measuring a time during which a medium, within which the link is defined, is busy; and

expressing the traffic density as a percentage of the measured time during which the medium is busy.

4. The method of claim 1, wherein resizing the first contention window comprises doubling a length of the first contention window only on second and subsequent transmission retries, but not on a first transmission retry.

5. The method of claim 1, additionally comprising:

determining a priority of each of a plurality of applications on the node including the first and second applications; and

associating contention window sizes with each of the plurality of applications based on the priority of each of the plurality of applications;

wherein the first size of the first contention window is one of the contention window sizes associated with one of the applications.

6. The method of claim 1, additionally comprising:

sizing the first contention window to the first size based on a calculation enforcing a plurality of bandwidth constraints on a plurality of applications on the node, respectively.

7. The method of claim 1, additionally comprising:

sizing the first contention window to the first size based on a calculation giving more bandwidth to the first application than the second application.

8. A node of a network, comprising:

one or more processors;

memory, in communication with the one or more processors;

a plurality of applications, defined in the memory;

a traffic load manager, defined in the memory and executable by the one or more processors, wherein the traffic load manager is configured to:

determine a priority of each of the plurality of applications; and

associate each of the plurality of applications defined in the memory of the node with one of a plurality of contention windows, the associating performed according to a priority of each application, wherein different applications from among the plurality of applications that have different priority have contention windows of different sizes;

a transmission retry manager, defined in the memory and executable by the one or more processors, to resize a contention window, the contention window from among the plurality of contention windows and associated with an application from among the plurality of applications, according to inputs including a quality metric and a traffic density over a portion of the network, wherein resizing the contention window comprises:

distinguishing between transmission failures due to traffic collisions and link quality;

resizing the contention window if transmission failures are due to traffic collisions;

reducing a short retry count maximum and reducing a long retry count maximum if the quality metric is below a first threshold and the traffic density exceeds a second threshold; and

informing an upper layer of a transmission protocol of the quality metric; and

a packet transmitter, defined in the memory and executable by the one or more processors, to transmit the packet to a second node.

9. A method of operating a node of a network, comprising:

associating a first contention window having a first size with a first application operating on the node;

associating a second contention window having a second size with a second application operating on the node, wherein the first size is different from the second size, and wherein the first and second contention windows are based on priorities of the first and second applications;

obtaining a quality metric of a link between the node and a destination node of a packet to be transmitted by the first application and a packet to be transmitted from the second application;

checking a level of traffic density of a portion of the network including the link;

resizing the first and/or second contention windows based on input including the quality metric of the link and the level of traffic density, the resizing comprising:

distinguishing between transmission failures due to the level of traffic density and the quality metric;

resizing the first contention window if transmission failures are due to the level of traffic density;

reducing a short retry count maximum and reducing a long retry count maximum if the quality metric is below a first threshold and the level of traffic density exceeds a second threshold; and

informing an upper layer of a transmission protocol of the quality metric; and

transmitting the packet from the first application and the packet from the second application after the resized first and/or second contention windows elapse, respectively.

10. The method of claim 9, additionally comprising:

determining which of a plurality of applications operating on the node, including the first and second applications, created data to be transmitted; and

selecting the first and/or second contention window based at least in part on the determination.

11. The method of claim 9, additionally comprising:

prioritizing a plurality of applications running on the node; and

associating each of the plurality of applications, including the first and second applications, with a contention window sized at least in part based on priority of the application.

12. The method of claim 9, additionally comprising:

utilizing a plurality of differently sized contention windows in association with a plurality of different applications operating on the node, respectively, the plurality of different applications including the first and second applications, wherein the plurality of different applications have a plurality of different priorities, respectively.

13. The method of claim 9, wherein sizing the contention window comprises:

sizing a plurality of contention windows differently for transmission of data created by a plurality of different applications, respectively, the plurality of different applications including the first and second applications.

CLAIMS

What is claimed is:

1. A method, comprising:

transmitting a packet from a node in a network, the transmission following elapsing of a contention window;

waiting a timeout period without receipt of an acknowledgement of the packet;

obtaining a quality metric of a link between the node and a destination node of the transmitted packet;

checking a traffic density of a portion of the network including the link;

resizing the contention window from an original size, the resizing based on input including the quality metric of the link and the traffic density; and

retransmitting the packet after the resized contention window elapses.

2. The method of claim 1, wherein obtaining the quality metric of the link comprises:

operating an expected transmission time algorithm on the node; and

calculating the quality metric based on the expected transmission time algorithm.

3. The method of claim 1, wherein checking the traffic density comprises:

measuring a time during which a medium, within which the link is defined, is busy; and

expressing the traffic density as a percentage of the measured time with respect to a sliding window.

4. The method of claim 1, wherein resizing the contention window comprises:

if the quality metric of the link is good and the traffic density is low, then maintaining the contention window at the original size;

if the quality metric of the link is good and the traffic density is high, then doubling a length of the contention window;

if the quality metric of the link is bad and the traffic density is low, then maintaining the contention window at the original size; and

if the quality metric of the link is bad and the traffic density is high, then doubling the length of the contention window.

5. The method of claim 4, wherein if the quality metric of the link is bad and the traffic density is high, then doubling a length of the contention window only on second and subsequent retries, but not on a first retry.

6. The method of claim 4, wherein resizing the contention window comprises:

if the quality metric of the link is bad and the traffic density is high, then reducing a short retry count maximum and reducing a long retry count maximum; and

informing an upper layer of a transmission protocol that the quality metric of the link is bad.

7. The method of claim 1, additionally comprising:

sizing the contention window to the original size based at least in part on which of at least two applications created data in the packet.

8. The method of claim 1, additionally comprising:

determining a relative priority of each of at least two applications on the node; and

associating contention window sizes with each of the at least two applications based on the relative priority of each of the at least two applications;

wherein the original size of the contention window is one of the contention window sizes associated with one of the applications.

9. The method of claim 1, additionally comprising:

sizing the contention window to the original size based on a calculation enforcing a plurality of bandwidth constraints on a plurality of applications on the node, respectively.

10. The method of claim 1, additionally comprising:

sizing the contention window to the original size based on a calculation giving more bandwidth to a first application than a second application.

11. A node of a network, comprising:

one or more processors;

memory, in communication with the one or more processors;

a plurality of applications, defined in the memory;

a traffic load manager, defined in the memory and executable by the one or more processors to calculate an original size of a contention window based at least in part on which of the plurality of applications created data included within a packet;

a transmission retry manager, defined in the memory and executable by the one or more processors to resize the contention window, according to inputs including representations of quality of a link and traffic density over a portion of the network; and

a packet transmitter, defined in the memory and configured to transmit the packet to a second node.

12. The node of claim 11, wherein the traffic load manager is configured to:

determine a relative priority of each of the plurality of applications;

associate contention window sizes with each of the plurality of applications based on the determined relative priority; and

size the contention window to the original size using as input at least one of the associated contention windows sizes.

13. The node of claim 11, wherein the transmission retry manager is configured to:

obtain a quality metric of a link between the node and at least one other node in the network;

check a level of traffic density of a portion of the network including the link;

if the quality metric of the link is good and the level of traffic density is low, then leaving the contention window at the original size;

if the quality metric of the link is good and the level of traffic density is high, then doubling a length of the contention window;

if the quality metric of the link is bad and the level of traffic density is low, then leaving the contention window at the original size; and

if the quality metric of the link is bad and the level of traffic density is high, then doubling the length of the contention window.

14. The node of claim 13, wherein the transmission retry manager is configured to:

if the quality metric of the link is bad and the level of traffic density is high, then:

reduce a short retry count maximum and reduce a long retry count maximum; and

inform an upper layer of a protocol used in the network that the quality metric of the link is bad.

15. A method of operating a node of a network, comprising:

sizing a contention window on the node;

obtaining a quality metric of a link between the node and a destination node of the packet to be transmitted;

checking a level of traffic density of a portion of the network including the link;

resizing the contention window based on input including the quality metric of the link and the level of traffic density; and

transmitting a packet after the resized contention window.

16. The method of claim 15, wherein sizing the contention window comprises:

sizing the contention window based at least in part on which of at least two applications on the node created data to be sent in the packet.

17. The method of claim 15, wherein sizing the contention window comprises:

associating each of several applications running on the node with a size of a contention window;

determining which of the several applications created data in the packet; and

sizing the contention window on the node based at least in part on the determination.

18. The method of claim 15, wherein sizing the contention window comprises:

prioritizing each of several applications running on the node; and

associating each of the several applications with a size of a contention window based at least in part on the priority of the application;

wherein sizing the contention window on the node is based at least in part on the prioritizing and the associating.

19. The method of claim 15, wherein sizing the contention window comprises:

prioritizing each of multiple applications running on the node based on relative importance and quantity of data to be transmitted over the network;

associating each of the several applications with a size of a contention window based on the priority of the respective application; and

utilizing a sequence of differently sized contention windows to send a sequence of different packets, respectively, based on the associations.

20. The method of claim 15, wherein sizing the contention window comprises:

sizing the contention window differently for transmission of each of a plurality of different packets, the plurality of different packets having data created by a plurality of different applications, respectively.