11-1-2008

# The Evolution of Internet Routing Metrics and Cost Calculations

Nigel Houlden
*Glyndwr University*, n.houlden@glyndwr.ac.uk

Vic Grout
*Glyndwr University*, v.grout@glyndwr.ac.uk

Rich Picking
*Glyndwr University, Wrexham*, r.picking@glyndwr.ac.uk



Recommended Citation
Houlden, N., Grout, V. & Picking, R., "The Evolution of Internet Routing Metrics and Cost Calculations", Proceedings of the Fourth Collaborative Research Symposium on Security, E-learning, Internet and Networking (SEIN 2008), 5-9 November 2008, Glyndwr University, Wrexham, UK, pp145-155.

# The Evolution of Internet Routing Metrics and Cost Calculations

**Abstract**
In this paper we examine how metrics impact route selection, looking at which metric(s) are selected and how these metrics are used to calculated cost. Examining multiple and dynamic metrics currently in use and looking toward a proposal of agent carrying dynamic metric information across an autonomous system.

**Keywords**
Routing protocols, routing agents, metrics, costs

**Disciplines**
Computer and Systems Architecture | Digital Communications and Networking | Hardware Systems | Systems and Communications

**Comments**
This paper was presented at the Fourth Collaborative Research Symposium on Security, E-learning, Internet and Networking (SEIN 2008), 5-9 November 2008, which was held at Glyndwr University, Wrexham, UK. It was published by the University of Plymouth, and the symposium proceedings are available at http://www.cscan.org

# The Evolution of Internet Routing Metrics and Cost Calculations

Nigel Houlden, Vic Grout and Rich Picking

Centre for Applied Internet Research (CAIR), Glyndŵr University, Wrexham, UK
{n.houlden|v.grout|r.picking}@glyndwr.ac.uk

## Abstract

In this paper we examine how metrics impact route selection, looking at which metric(s) are selected and how these metrics are used to calculated cost. Examining multiple and dynamic metrics currently in use and looking toward a proposal of agent carrying dynamic metric information across an autonomous system.

## Keywords

Routing protocols, routing agents, metrics

## 1. Introduction

As computer networks expand and more data flows over them the need to ensure that routing is optimized is greater than ever. Most current routing strategies do not work for the good of the network; they actively compete over links, relying upon cost functions calculating the lowest cost.

How the lowest cost is calculate is a problem, a cost is calculated based upon some metric and then, in most cases, simple summation of link costs to produce a path cost. The question of what metrics are used or should be used varies from protocol to protocol, some are simplistic and other appear to be more complicated taking account of multiple and claimed dynamic metrics.

(Ali, Mouftah, El-Sawi, 1997) state that the exchange of topology state information helps in keeping the network nodes updated on the network status and hence making the correct routing decision, but does it, if the information is not truly dynamic (real time) then is it any use?

Some of the most common protocols are RIP (Routing Information Protocol), OSPF (Open Shortest Path First) and Cisco's EIGRP (Enhance Interior Routing Protocol) the successor to IGRP (Interior routing protocol).

This paper will examine the effects of using single or multiple metrics for routing calculation and whether true dynamic metrics are used, in section 2 an outline of how current protocols calculate cost is shown, in section 3 more detail is given to Cisco's

EIGRP and how its metrics affect the routing decision when they are implemented and if they are truly dynamic and finally the conclusions are in section 4.

## 2. Current protocols

There are currently many protocols that can be implemented in order to populate routing tables. In this section four will be examined RIP, OSPF, IGRP and EIGRP.

### 2.1. Routing Information Protocol (RIP)

RIP is simplistic in its calculation using hop count as a metric; this takes the least number of hops from source to destination and uses this as the lowest cost. However this can actually be detrimental to a network, links with a low bandwidth can be utilised over links with a higher bandwidth, this is not a particularly sophisticated method and could easily cause delays.

### 2.2. Open Shortest Path First (OSPF)

Open shortest path first utilises bandwidth in its cost calculation by simply calculating the inverse of it, $c = \frac{10^8}{b}$ , although this is still simplistic in its approach it is an improvement over RIP as now links with greater bandwidth will be used. OSPF uses Dijkstra's algorithm to determine the distances (costs) between a given node and all other nodes in a graph. The algorithm begins at a specific node and extends outward within the graph, until all nodes have been reached, Dijkstra's algorithm stores the total cost from a source node to the current node.

The cost of each link is then added to calculate total cost

$$C_T = \sum_{i=1}^{n} c_i$$

However this has its limitations, if all that is measured is the bandwidth how can the current state of a link be determined?

### 2.3. Interior Gateway Routing Protocol (IGRP)

Cisco's IGRP is a proprietary protocol and is worth discussing before examining EIGRP, the enhanced version. The calculation for IGRP is given in 1

$$c = (k_1 b + \frac{k_2 b}{l - 256} + k_3 d)(\frac{k_5}{r + k_4}) \qquad (1)$$

However this is not the default calculation, by default the k values k1 and k3 are set to 1 and the others set to 0.

This gives just Bandwidth and Delay for the calculation, but again the calculation is not that simple, some adjustments have to be made (Cisco 2005a) for the bandwidth, find the smallest of all the bandwidths in Kbps from outgoing interfaces and divide 10,000,000 by that number. (The bandwidth is scaled by 10,000,000 in kilobits per second.) for the delay, add all of the delays (in microseconds) from the outgoing interfaces and divide this number by 10. (The delay is in tenths of microseconds.) Therefore the default calculation for IGRP cost is

$$c = \left( \frac{1x10^7}{b} + \frac{\sum_{i=1}^{n} d_i}{10} \right)$$

**2.4. Enhanced Interior Gateway Routing Protocol (EIGRP)**

EIGRP is a more advanced protocol; providing a more realistic (although still imperfect) end-to-end cost calculation (Houlden et al 2006). Again it is Cisco proprietary protocol and takes in to account the metrics, bandwidth *b*, delay *d*, load *l* and reliability *r*, the calculation for each link cost is shown in 2, the same as IGRP

$$c = (k_1 b + \frac{k_2 b}{l - 256} + k_3 d)(\frac{k_5}{r + k_4}) \qquad (2)$$

Each of the metrics can be excluded, like IGRP, by setting the k values to 0 or any value to emphases  a particular metric and influence a route , as with IGRP by default $k_1$ and $k_3$ = 1 and $k_2$, $k_4$ and $k_5$ = 0, (Cisco, 2005b) notes that (2) only applies if $k_5$ is not 0. So the actual cost calculation is

$$c = (k_1 b + k_3 d) \qquad (3)$$

If $k_5$ does not equal zero, an additional operation is performed

$$c = (c(\frac{k_5}{r + k_4})) \qquad (4)$$

However Cisco also states that the EIGRP cost has to be scaled by a factor of 256, this is for compatibility with IGRP. With IGRP the overall metric is 24 bit where EIGRP is a 32 bit, the 256 is effectively a scaling factor allowing redistribution between the two protocols by either dividing or multiplying by 256 giving

$$c = \left( \frac{1x10^7}{b} + \frac{\sum_{i=1}^{n} d_i}{10} \right) 256$$

There are issues with delay also, Cisco gives a table of values for a particular media type although this can be manually configure delay by default is no more than a function of bandwidth.

| Media Type | Delay | Bandwidth |
|---|---|---|
| Satellite | 5120 (2 seconds) | 5120 (500 megabits) |
| Ethernet | 25600 (1 milliseconds [ms]) | 256000 (10 megabits) |
| 1.544 Mbps | 512000 (20,000 ms) | 1,657,856 bits |

**Table 1: Comparing delays and bandwidths**

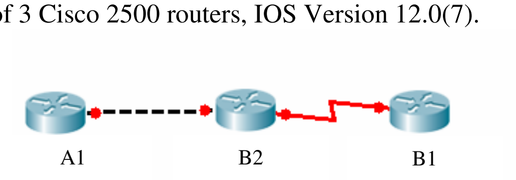
Delay is stored in a 32-bit field, in increments of 39.1 nanoseconds, ranging from 1 which equals 39.1 ns, to a hex value of FFFFFFFF, 4,294967040 ns.

By default the cost calculation is fairly simplistic using bandwidth and delay and some scaling.

However this still leaves reliability and load, reliability is represented as a value of 255/255 for 100% reliable and load is a fraction of 255, 255 indicating the link is completely saturated. Load and reliability are only used in the calculation of cost if the *k* values are set, *k*2 and *k*5 respectively.

## 3. EIGRP Reliability

In order to test the use of reliability in EIGRP a small network was established consisting of 3 Cisco 2500 routers, IOS Version 12.0(7).




The routers were configured as follows:

A1 Ethernet interface 0 IP address 10.0.0.2/24, B2 Ethernet interface 0 IP address 10.0.0.1/24, Serial interface 0 IP address 192.168.0.1/24 and router B1 Serial interface 0 IP address 192.168.0.2/24. Each router had EIGRP enabled within the same Autonomous System (AS), this was the default configuration so the cost calculation is

$$c = \left( \frac{1x10^7}{b} + \frac{\sum_{i=1}^{n} d_i}{10} \right) 256$$

Firstly the value of reliability and the cost was observed from B1 to A1.

```
B1#sh ip route 10.0.0.2
Routing entry for 10.0.0.0/8
  Known via "eigrp 1", distance 90, metric 2195456, type internal
  Redistributing via eigrp 1
  Last update from 192.168.0.1 on Serial0, 00:02:12 ago
  Routing Descriptor Blocks:
  * 192.168.0.1, from 192.168.0.1, 00:02:12 ago, via Serial0
      Route metric is 2195456, traffic share count is 1
      Total delay is 21000 microseconds, minimum bandwidth is 1544 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
```

The reliability was tested by disconnecting the link between A1 and B2 and viewing the result on the metric the value of reliability starts to drop and continues to do so, this was left and then the cable reconnected. The route from B1 to A1 was examined again.

```
B1#sh ip route 10.0.0.1
Routing entry for 10.0.0.0/8
  Known via "eigrp 1", distance 90, metric 2195456, type internal
  Redistributing via eigrp 1
  Last update from 192.168.0.1 on Serial0, 00:00:00 ago
  Routing Descriptor Blocks:
  * 192.168.0.1, from 192.168.0.1, 00:00:00 ago, via Serial0
      Route metric is 2195456, traffic share count is 1
      Total delay is 21000 microseconds, minimum bandwidth is 1544 Kbit
      Reliability 211/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
```

The reliability metric has changed value but as expected the cost of the link has remained constant, showing that the default calculation does not include reliability even though the metric on the interface was seen to alter.

The routers were then reset and the same configuration applied, this time the metric weights of EIGRP where altered to 1. By altering the *k* values to include reliability the same experiment was carried out.

```
B1#SH IP ROUTE 10.0.0.2
Routing entry for 10.0.0.0/8
  Known via "eigrp 1", distance 90, metric 8601, type internal
  Redistributing via eigrp 1
  Last update from 192.168.0.1 on Serial0, 00:02:35 ago
  Routing Descriptor Blocks:
  * 192.168.0.1, from 192.168.0.1, 00:02:35 ago, via Serial0
      Route metric is 8601, traffic share count is 1
      Total delay is 21000 microseconds, minimum bandwidth is 1544 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
```

Again the link was disconnected and the value of reliability starts to drop, this was left and then the cable reconnected. The route from B1 to A1 was examined again.

```
B1#sh ip route 10.0.0.1
Routing entry for 10.0.0.0/8
  Known via "eigrp 1", distance 90, metric 17069, type internal
  Redistributing via eigrp 1
  Last update from 192.168.0.1 on Serial0, 00:00:00 ago
  Routing Descriptor Blocks:
  * 192.168.0.1, from 192.168.0.1, 00:00:00 ago, via Serial0
      Route metric is 17069, traffic share count is 1
      Total delay is 21000 microseconds, minimum bandwidth is 1544 Kbit
      Reliability 128/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
```

The reliability has dropped to 128/256 and the cost has increased, making this link less desirable if an alternative were available. However even after 24 hours the route cost had not returned to its original costing and the route

This gives a clear result that reliability is dynamic and changes the cost as the link reliability alters. However this begs the question of what is the lowest limit that reliability can go to and how is the cost affected. Figure 1 shows the results of the reliability on A1 Ethernet interface being recorded every 10 seconds.

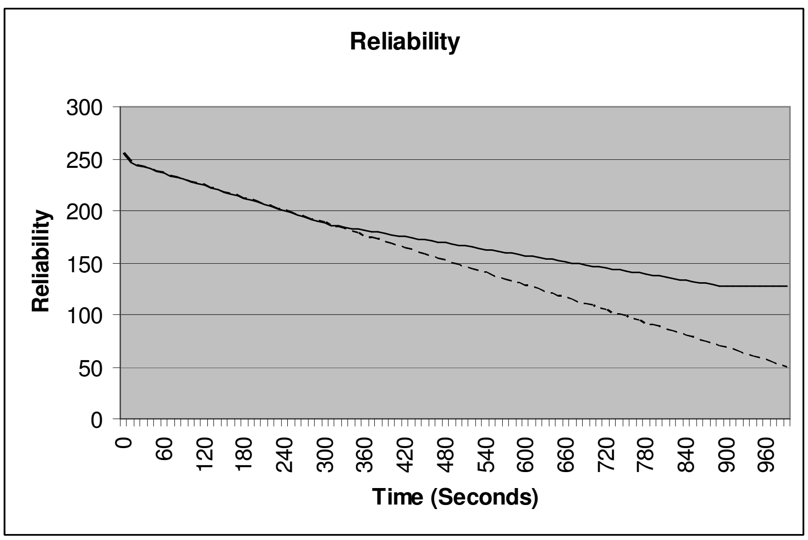



**Figure 1: Watching the reliability fall**

The solid line shows the reliability falling at a steady rate of 2 units every 10 seconds, the dashed line is the predicted result for this rate of change, however at 320 seconds the rate changes to one unit every 10 seconds but only continues this up to 890 seconds where the value of reliability reaches 128/255. At this point the rate halts and remains at this level, the cable was left disconnect for 66 hours and the reliability remained at 128/255.

## 4. The Future

Even though the EIGRP metrics can be dynamic and alter the cost, it appears that once reliability has been changed it does not recover alone, a router reset is required. However this does show that changing metrics can affect cost and in turn alter routing decisions, but are the metrics in EIGRP the correct ones to use? And if they are is the calculation of cost the correct one?

There is a current trend to look toward queue delay as a metric for cost calculation (Vijayalakshmi and Radhakrishnan), this is not a new concept (Dorigo and Stutzle 2004) designed AntNet, a routing algorithm based upon Simple Ant Colony Optimization (SACO). AntNet uses the time elapsed since the launching time to the arrival time at the *i*-th node.

The principles of AntNet are laudable, essentially it has virtual ants, traversing specific path eventually relaying delay information to each node in the path. This is carried out by two types of ant, a forward and a backward ant, the algorithm has two main phases, solution construction and data structure update.

In the solution construction stage a forward ant is launched from a source node to a destination node $F_{s \to d}$ this is treated as a normal data packet by the network and is subject to delay, which is recorded. The destination node is selected according to data traffic patterns generated by local work load, $p_{sd} = \frac{f_{sd}}{\sum_{i=1}^{n} f_{si}}$ calculates the probability of a node creating a forward ant to node d as the destination, where $f_{sd}$ is a measure in bits or number of packets of the data flow $s \to d$ .

At each node *i* decisions are made in order to prevent loops, when an ant reaches the destination node *d* the agent $F_{s \to d}$ generates a new agent $B_{d \to s}$ and transfers all of its memory to it. The backward ant then takes the same path back to source but in the other direction, as the ant reaches node *i* in the path the routing information is now updated.

The AntNet system uses a pheromone matrix $\mathrm{T}_i$ at each node that is updated as the backward ant arrives. The elements of $\mathrm{T}_i$ are $\tau_{ijd}$'$s$, these indicate the learned desirability for an ant in node *i* to move to node *j* with a destination of *d.* Since the problem consists of the solution of many $\frac{n(n-1)}{2}$ the AntNet pheromones have three indices, this means an ant can from node *i* have any of the remaining *n-1* nodes as a destination.

Using queue delay, as AntNet does, has advantages; the greater the bandwidth of a link the queue delay should be shorter, as traffic on that link grows the delay increases. By using one metric, queue delay, the bandwidth, the delay and the load can be taken in to consideration, giving the simple cost calculation of

$$c = \sum_{i=1}^{n} q_i$$

However this should still be expanded, with EIGRP the reliability can be taken in to consideration, regardless of how short or long a queue maybe, if the link is unreliable it needs to be taken in to consideration.

$$c = \left( \sum_{i=1}^{n} q_i \sum_{j=1}^{n} r_j \right)$$

The cost calculation now reflects that if a link has problems with reliability it would become less attractive. Even so this calculation should still be expanded to include bandwidth, when a router is first initialized the queue will be non existent, and a link should be one hundred percent reliable.

$$c = \left( \sum_{i=1}^{n} q_i \sum_{j=1}^{n} r_j \sum_{x=1}^{n} b_x \right)$$

So it would seem that by using three metrics the cost of a link can be calculated and any changes accounted for, but surely queue delay and reliability should factor more heavily then bandwidth, this is explored by (Grout et al 2006), giving a matrix for summation, product, minimum or maximum, so this can be reflected in the cost calculation by finding the product of the delay and the product of reliability

$$c = \left( \prod_{i=1}^{n} q_i \prod_{j=1}^{n} r_j \sum_{x=1}^{n} b_x \right)$$

thus placing more emphasis on reliability and queue delay, however some factoring should be added to ensure sensible values

$$c = \left( \left( \prod_{i=1}^{n} q_i \right) \left( \frac{\prod_{j=1}^{n} r_j}{5} \right) \left( \frac{1}{\sum_{x=1}^{n} b_x} \right) \right)$$

The product of queue delay is calculated along a path, the more delay on each link the greater the factor of delay should be calculated, reliability is calculated in percent, if the link is 100% reliable a value of one is used else the unreliable percentile is used, 90% reliable gives 10% unreliable and would be calculated as interface downtime compared to router uptime.

Using these metrics for a calculating the cost of a link would be beneficial, but it would be more beneficial if we could use these metrics to calculate path or domain route costs. By using agents similar to AntNet the values of links could be carried from router to router.

An Agent would be launched from a node *i* along path $a \Rightarrow b$ via node *j*, path $a \Rightarrow b$ would be selected randomly by node *i*. At each node *j* the agent would update the node with metrics gathered from link $i \rightarrow j$, based upon current routing information the agent would be transmitted on the 'best' interface to reach the destination. Upon reaching the next node in the path the metrics for the traversed links would be used to update the nodes routing strategy, this system would continue until the destination node was reached. When the agent reaches the destination the final node in the path updates it routing strategy.

At this stage one of two things will occur, firstly if the destination node has an alternative path to the source node the agent will be re-transmitted along this route with the information that was gathered along the original path *a,b* and gather new data for this alternative route, updating each node along the way and eventually arriving at the source with data about both routes, at this point the agent would be destroyed. However if at the destination no alternative path back is known then the agent is destroyed and the destination node selects a new path $a \Rightarrow b$ and creates a new agent.

This may seem a selfish act since the source node receives no update as to this path, however the routing strategy should not be centered around a single node discovering the best path, a so called selfish node (Omrani and Fallah, 2008), it should be about nodes exchanging data about links for the whole of the network. As each destination node either sends the agent back via an alterative route or creates a new agent the network is learning not just about good paths but about the current state of the network, giving a centralized view through a distributed method, allowing each node to work for the good of the network and not just its self.

Having this system will benefit quality of service, diverting traffic from over subscribed links to links that would have not normally have been selected because bandwidth was too small, an unused link no matter how low the bandwidth is better than a high bandwidth one that has to drop packets due to high traffic. The data carried by the agents could also be used as part of a reporting mechanism, if a link is over subscribed and no alternative is available this information can be feed in to a reporting mechanism, allowing the administrator to decide on a course of action, upgrade a link or install a new link allowing an alternative path?

## 5. Conclusion

Being able to alter the values of metrics is obviously advantageous; influencing route choice when implemented by a skilled administrator can only enhance a network, but if a network could do this automatically, dynamically in order to change routes as the preferred route becomes altered in someway would undoubtedly make networks more efficient.

From the reliability test that has been shown, cost can be changed dynamically and therefore altering routing tables, if this information could be propagated across a domain, so that a picture of the state of the network could be made and routes updated because of this there is a chance of true optimization.

However are these dynamic values enough, or should they be viewed in a more detail way, if a cost changed by 50% but was still the lowest cost should that link still be used or should traffic be diverted through an alternative path until such a time that the cost falls back into a reasonable parameter. As was shown in section 3, Cisco add the delays across a path, is simple addition the correct way to calculate total delay?

In the future networks are going to become more complicated, faster and harder to administer. If a self correcting, dynamic network routing protocol could be developed this would make complicated networks more manageable. Some work has already been carried out in this area, ACO (Houlden et el 2005) proposed an agent based system of ants, moving through an Autonomous System carrying information about the state of links, this work will continue, as described in the previous section having an agent based system would work for the entire network not just individual nodes.

It is with this in mind that the next steps are to test a number of protocols, such as AntNet and two heuristic, distributed (router-based) P4a algorithms given in (Grout 2004b) and (Grout et al 2004c) which seek to minimize

$$K^{x}_{<network>} = \sum_{a \in N} \sum_{i \in N} \sum_{j \in N} c(\sum_{b \in N} x^{ab}_{ij} t_{ab,} m_{ij)})$$

against EIGRP and OSPF using the OPNET network simulator, from this a new algorithm for an agent based system will be developed and tested.

## 6. References


Ali, I.A.; Mouftah, H.T.; El-Sawi, A.H. A dynamic routing protocol for broadband networks**,** Computers and Communications, 1997. Proceedings., Second IEEE Symposium 1-3 Jul 1997 Page(s):495 - 500

Cisco (2005a) "IGRP Metric, Document ID: 13678"

http://www.cisco.com/en/US/tech/tk365/technologies_tech_note09186a008009405c.shtml (accessed 18th June 2008)

Cisco (2005b) "IP Routing Protocols Commands: K through M", http://www.cisco.com/en/US/docs/ios/12_3/iproute/command/reference/ip2_k1g.html#wp1040267 (accessed 23rd May 2008)

Dorigo M and Stutzle T, 2004, Ant Colony Optimization, MIT press, Massachusetts, ISBN 0-262-0419-3

Grout, V. & Houlden, N. Principles of Optimal Interior Routing, Proceedings of the 1st International Conference on Internet Technologies and Applications (ITA 05), Wrexham, North Wales, UK, 7th-9th September 2005, pp274-281.

Grout, V.2004a, Scope and Resolution in Local Search Optimisation, Proceedings *of Combinatorial Optimisation 2004*, Lancaster University, UK, 2004

Grout, V., 2004b, "A Self-Partitioning Link-State Routing Protocol", Proceedings of IEE/IEEE ICN'04, Gosier, Guadeloupe, French Caribbean, 1st-4th March

Houlden, N., Grout, V., McGinn, J. & Davies, J. Extended End-to-End Cost Metrics for Improved Dynamic Route Calculation, Proceedings of the 6th International Network Conference (INC 2006), University of Plymouth, 11-14 July 2006, pp89-96.

Omrani A, Fallah M, 2008 "A Game Theoretic Cooperation Stimulus Routing Protocol for MANETs" IAENG International Journal of Computer Science, 35:1 2008.

Vijayalakshmi K and Radhakrishnan S, 2007 "Dynamic Routing to Multiple Destinations in IP Networks using Hybrid Genetic Algorithm (DRHGA)" International Journal of Information Technology Volume 4 Number 1, 2007