Financial Derivatives for Computer Network Capacity Markets with Quality-of-Service Guarantees

Petter Pettersson
pp@kth.se

February 2003
SICS Technical Report T2003:03

Keywords Networking and Internet Architecture.

Abstract

Five network services that meet the needs of new Internet applications are formulated as derivatives on the spot price of the capacity in network connections. The derivatives are priced using the Black-Scholes model. The services suggested are: a multicast service for one-to-many connections, a video on-demand service, a service that gives the user perceived higher quality for many applications, a service that allows the requested capacity to vary with time and a service that does not specify the exact time of delivery of capacity.
Acknowledgement

This work was carried out at SICS, Swedish Institute of Computer Science within the framework of project AMRAM. I thank Erik Aurell and Lars Rasmusson for help and direction.
1 Introduction 5
  1.1 Spot markets for router capacity .......................... 5
  1.2 Quality of service ........................................ 5
    1.2.1 Network capacity ..................................... 6
  1.3 Structure of this thesis .................................... 6

2 Model 7
  2.1 Network architecture ...................................... 7
  2.2 Risk-neutral valuation and the Black-Scholes model ....... 9
  2.3 Rainbow options ......................................... 10
    2.3.1 Several underlying assets ............................ 11
    2.3.2 The $n$-dimensional Girsanov transform of $\mathbb{E}^0[S_g(S)]$ ... 11
  2.4 Definitions .................................................. 11
    2.4.1 The cost of resources ................................ 12
    2.4.2 The send fee ......................................... 12

3 New Internet applications 14
  3.1 Auditory applications ...................................... 14
  3.2 Video-based applications .................................. 15
  3.3 Remote control of instruments ............................. 16
  3.4 Grid computing ............................................. 16
  3.5 Virtual Environments ...................................... 17

4 Network services 17
  4.1 Reliable multicast ......................................... 17
  4.2 Video on-demand .......................................... 21
  4.3 Internet Capacity Boost .................................. 22
  4.4 Time-dependent capacity .................................. 23
  4.5 Traffic sometime during the night ........................ 24

5 Discussion 27

A Network derivatives 27
  A.1 Pricing example – Network forward ....................... 28
  A.2 Pricing example – Network cash-or-nothing option ....... 29

B Basket options 29
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>B.1 Valuation of basket options using Edgeworth-series expansion</td>
<td>29</td>
</tr>
<tr>
<td>B.2 Sum of two log-normal variables</td>
<td>31</td>
</tr>
<tr>
<td>B.3 Distribution expansion</td>
<td>32</td>
</tr>
</tbody>
</table>

**Index** 34

**References** 35
1 Introduction

The usefulness of a computer is greatly increased when it is connected to other computers by a network. A network of networks is called an internet and the largest internet in the world is called the Internet. This thesis is based on the idea that the owners of the networks that are part of the Internet should be able to sell their spare capacity on a spot market to individual users. This will enable new Internet applications that have high capacity requirements. The capacity rights can be traded in the form of services designed to provide capacity for the applications. The services are priced as financial derivatives of the underlying asset network capacity.

1.1 Spot markets for router capacity

The most common way of handling data streams through network routers is to use best-effort routing. When best-effort routing is used all streams of network traffic experience equal loss. There are however many applications that may not be accommodated by the best-effort Internet service model. There are some data packets, such as audio/video streams, that have deadlines. Over-crowding, or congestion, can cause unacceptable delays due to packet losses and retransmission if best-effort routing is used. As network routers become congested some users will thus want to reserve network capacity in the routers. When users are able to reserve capacity the network is said to be able to provide guaranteed quality of service, or QoS.

It is desirable that a reservation of capacity does not block other reservations and that the reservation scheme doesn’t require extensive negotiation. Trading router capacity in spot markets is a way to meet these requirements. In this market users buy or sell router capacity depending on their needs. As prices increases with demand alternative paths in the network become competitive and users tend to move their bandwidth usage away from congested routers.

The working hypothesis is that a suitable number of contingent claims, or derivatives, on router capacity will improve the efficiency of the router capacity market (see Rasmusson et al.[20]). The objective of this thesis is to design a number of services that are priced as financial derivatives. We assume that derivative prices are functions of current market prices and the statistical model. The model of price dynamics in a spot market used in this thesis is the one proposed by Rasmusson and Aurell[19]. Rasmusson has provided some useful theorems under the assumption that prices are log-normal[16] and a way of evaluating the CDF for weighted sums of correlated log-normal random variables[17].

1.2 Quality of service

Network Quality-of-Service can be described using different metrics. Networkers and application developers have different perspectives and do not measure quality in the same way[12]. Application performance is expressed in terms that focus on user-perceivable effects. A user may not perceive e.g. data loss in the network as loss of audio clarity. This means that data loss on the network level is not the same as from data loss on the application level.

Using the network perspective these are some of the most important metrics
that characterize performance:

- **Bandwidth.** The amount of data in bits-per-second (bps) that can be sent through a given communications circuit.

- **Delay.** The time it takes for data units to be carried by the network to the destination.

- **Delay variation.** This is often due to buffering on routers during periods of increased traffic.

- **Packet loss.** A packet is the unit of data sent across a network. Lost packets are usually a result of congestion on the network.

- **Loss pattern.** Periods and other patterns can be helpful when designing applications to handle packet loss.

There are other also other factors, such as re-ordering and security, that can be regarded as QoS metrics. One word that is often used to describe network quality is **reliability.** It is not completely clear what this means, but the interpretation advocated here is that it summarizes the effect of all the QoS metrics except bandwidth and delay. Delay variation, packet loss loss pattern, re-ordering and security are thus all factors that affect reliability.

When describing the guaranteed QoS sold on spot markets we will however simply use the metric **network capacity**:

### 1.2.1 Network capacity

In many situations it is convenient to use only one variable to describe Quality-of-Service – network capacity. Real network capacity is dependent in a complex way on several network quality factors, such as bandwidth, delay, jitter, packet loss and loss pattern. How the quality is affected by these factors depend on the application. A service that is valuable to one user may be worthless to another user if e.g. the delays are too long, regardless of the bandwidth.

It is not feasible to allow users to specify every QoS factor each time a service contract is bought since this would make the price negotiation too complicated. The way of getting around this problem will depend on the applications. A possible solution is to interpret capacity as bandwidth with some guarantees regarding delay and packet loss. A user will be able to buy more bandwidth, but will perhaps not be able to decrease delays or increase the reliability. For more on the quality levels in the model used here, see 2.1 on the following page.

### 1.3 Structure of this thesis

In chapter 2 the network architecture and mathematical models used in this thesis is presented. Chapter 3 surveys some of the most important new Internet applications. In chapter 4 network services are suggested to meet the capacity needs of some of the applications. Chapter 5 concludes the thesis with a discussion of services.
# 2 Model

In this chapter the network and derivatives model used in this thesis are described. First, the network architecture is described. Then financial mathematical models are introduced and applied to the network capacity market. The main reason for introducing a send fee as a part of the network architecture is explained.

## 2.1 Network architecture

The network architecture is one of the most important concepts when discussing computer communications. Stallings describes the architecture as the structured set of subtasks that implements communication[22]. The TCP/IP reference model is one example of an network architecture. TCP/IP (Transmission Control Protocol over Internet Protocol.) is divided into four layers: Link, Network, Transport and Application. They can be thought of as a stack with the Application layer at the top. The Link layer (a k a: Host to network layer, Layer 2) enables the upper layers to communicate with the hardware. The Network layer (a k a Internet layer, Layer 3) figures out how to get the data to its destination. The reliability of the transmission is guaranteed by the Transport layer and the Application layer provides a user interface. Another example is the OSI (Open Systems Interconnection) seven-layers model that among other things defines a Physical layer, which is the physical medium used to transmit data.

Here follows a short description of Rasmusson’s network architecture. A complete description can be found in Rasmusson’s dissertation[18].

**Rasmusson’s architecture** This Internet consists of several subnetworks (figure 1 on the next page). In this architecture, the network traffic is switched, monitored and measured by neutral managers at exchange locations, or exchange points. The exchange locations are nodes in the graph that describe the network layout (figure 2 on the following page).

They connect subnetworks owned and run by network owners such as operators, companies and universities. The owners determine how much traffic that can be sent between the exchange locations at the edges of the subnetwork. This capacity is then announced as available and sold in shares at a market-place. The owners may change the flows through the subnetwork (figure 3 on the next page) as they want, as long as the traffic is delivered as promised.

The network capacity shares are sold in bundles called network services. Network users are persons, companies or automatic agents. An automatic agent is “a software component that acts on our behalf, with our authority, and that is intended to do so in our best interest.[10]” All users are assumed to be self-interested in the sense that they prefer better performance for themselves to better performance for someone else.

The network owners must be able to handle two traffic classes, providing different QoS levels:

1. **The First class.** First class traffic is guaranteed to be handled by the network. To send traffic in this class a user must pay for the reserved
Figure 1: The Internet consists of a large number of subnetworks. In this example there are five subnetworks connected by exchange locations. The users access to network capacity is controlled by access points. Figure by the author based on a figure in [18].

Figure 2: The exchange locations are nodes in the graph that describe the network layout. The users at the edge of the network have several different paths (solid black lines) to choose from.

Figure 3: A closer look at one of the subnetworks. The network owner of the subnetwork configures paths for reservable capacity between border routers. The configuration of internal routers and links need only be known by the network owner. Figure by the author based on a figure in [18].
capacity. Guaranteed Quality-of-Service, or guaranteed network capacity, is provided. Not only the amount of bandwidth, but also the level and character of other quality measures, such as delay and packet loss, is specified.

2. The Best-effort class. Users can send traffic in this class for free, but the class offers no guarantee at all. Packets may be thrown away when the network becomes congested.

By combining these two traffic classes in different ways one can implement any level of Quality-of-Service. The price of a service is affected only by the amount of traffic sent in the first class, since traffic in the best-effort class is free. The shares of first class traffic that are sold at the market specify the capacity and the entry and exit gateway addresses, and the exchange locations between which the share provides capacity. A share owner is guaranteed to send packets at the specified rate indefinitely. Capacity tokens are used to verify that the user has reserved capacity. The tokens are bit-strings with cryptographically signed contracts. The user pays not only for the shares, but also has to pay a per-second send fee (see 2.4.2 on page 12) to the network owner. The send fee is necessary to price network capacity derivatives.

Some of the exchange points are also access points that users connect to. The access points shape the traffic from the user so that it does not exceed the allowed amount anywhere along the path. Traffic that exceeds the allowed rate is marked as best-effort traffic. The access point admits traffic only if the user can send capacity tokens to prove that he owns sufficient shares. There will be one market for each kind of capacity, that is, each exchange node.

End-users may buy capacity on the spot markets, but it is assumed that most users will buy the capacity in the form of services. The services are financial derivatives of network capacity assets. The derivatives are priced and sold to the end-users by middle-men, or brokers. The next section describes the pricing model used to price derivatives of network capacity.

2.2 Risk-neutral valuation and the Black-Scholes model

Descriptions of the Black-Scholes[3] method of pricing options and other derivatives are given in e.g. Hull[8], Bingham-Kiesel[1] and Björk[2]. The Black-Scholes world under the probability measure $\mathbb{P}$ is

$$dS(t) = S(t) \left( \mu dt + \sigma dW(t) \right), \quad S(0) = S_0 \in (0, \infty)$$

$$dB(t) = rB(t)dt, \quad B(0) = 1$$

where $r$, $\mu$ and $\sigma$ are constant coefficients. The constant $r$ denotes the risk-free interest rate, $\sigma$ is the volatility. $W(t)$ is Brownian motion, a stochastic process with normal distribution such that $\mathbb{E}[W(t)] = 0$ and $\mathbb{E}[W(t)^2] = t$.

The measure $\mathbb{P}$ can be thought of as the ordinary world, a world where investors do care about risk. The constant $\mu$ is a measure of risk aversion by investors; a higher $\mu$ means a higher risk aversion. In order to price derivatives we move into a world where investors are risk-neutral. This means changing measure to
\( Q \), often called the equivalent martingale measure. The price we will obtain is, however, valid in all worlds. Under the probability measure \( Q \) the world is

\[
\begin{align*}
    dS(t) &= S(t) \left( rdt + \sigma dW^Q(t) \right), & S(0) = S_0 \in (0, \infty) \\
    dB(t) &= rB(t)dt, & B(0) = 1
\end{align*}
\]

where \( W^Q(t) \) is Brownian motion. Under this measure \( S(t)/B(t) \) is a martingale. Equation 3, a linear SDE, is known as geometric Brownian motion and has the (strong) solution

\[
    S(t) = S_0 \exp \left( (r - \sigma^2/2)t + \sigma W^Q(t) \right).
\]

In e.g. Hull[8] the Black-Scholes PDE is derived from equations 1 and 2 and a set of assumptions. The equation is

\[
\begin{align*}
    rf(t, S) &= \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 f}{\partial S^2} + rS \frac{\partial f}{\partial S} + \frac{\partial f}{\partial t} \in [0, T) \times \mathbb{R}, \\
    f(T, S) &= \Phi(S)
\end{align*}
\]

where \( S = S(t) \). Part of the Black-Scholes PDE (equation 5) is recognized as being the generator of geometric Brownian motion. Let \( f \in C^2_c(\mathbb{R}) \), i.e. \( f \) is a twice continuously differentiable function with compact support. The (infinitesimal) generator \( A \) of the geometric Brownian motion under \( Q \) is (see e.g. Øksendal[14])

\[
    Af(x) = \frac{1}{2} \sigma^2 x^2 f''(x) + rx f'(x).
\]

The term \( rf(t, S) \) in equation 5 represents the possibility of investors to invest in the risk-less asset \( B(t) \).

If the Black-Scholes equation is used with the Feynman-Kac formula we get a representation of \( f(T,x) \) which is the price of an attainable contingent claim \( \Phi(S(T)) \) at time \( t \)

\[
    f(t, S(t)) = e^{-rT} \mathbb{E}^Q[\Phi(S(T)) | F_t]
\]

This is the Black-Scholes version of the risk-neutral valuation formula. There are more general versions of this formula, where the numéraire (in the Black-Scholes case \( e^{-rT} \)) and the asset are substituted for more general ones. See e.g. Bingham-Kiesel[1].

### 2.3 Rainbow options

Options involving more than one risky asset are often referred to as rainbow options. One example of a rainbow option is the basket option, an option whose
payoff depends on the value of a portfolio of assets. Another example is options on the minimum or maximum of two or more assets. The network derivatives in this thesis are all rainbow options, since the payoff generally involve more than one risky asset. The cost of resources, defined by equation 10 and path send fee 12 are both examples of baskets of assets.

Rasmusson has suggested a method of pricing network capacity derivatives by using Monte Carlo simulation to evaluate the PDF of the asset basket[20]. Basket options can also be priced e.g. by using Edgeworth series expansion[9]. This is described briefly in the appendix of this thesis (appendix B.1 on page 29).

2.3.1 Several underlying assets

In Björk[2] the Black-Scholes model is generalized to the case where we have several risky assets apart from the risk-free asset. If there are \( n \) risky assets, the asset price vector is

\[
S(t) = [S_1(t) \ldots S_n(t)]^T
\]

The contingent claims are of the form \( \Phi(S(T)) \) where \( T \) is the fixed exercise time. The price vector is assumed to be driven by \( n \) independent Wiener processes. But, since it is reasonable to assume that the prices \( S_i \) are correlated, each price process is assumed to be dependent on all \( n \) Wiener processes. Under the objective probability measure \( \mathbb{P} \), the \( S \)-dynamics is given by

\[
dS_i(t) = \mu_i S_i(t) dt + S_i(t) \sum_{j=1}^{n} \sigma_{ij} dW_j(t), \quad S_i(0) = S_{0,i}
\]

where \( \mu_i \) and \( \sigma_{ij} \) are assumed to be known constants. The volatility matrix \( \Sigma = \{\sigma_{ij}\}_{i,j=1}^{n} \) is assumed to be nonsingular. We also have the risk free asset defined by equation 2 on page 9. The volatilities are the same under \( \mathbb{Q} \) as under \( \mathbb{P} \) and the risk-neutral valuation formula (equation 6 on the page before) still holds.

2.3.2 The \( n \)-dimensional Girsanov transform of \( E^Q[Sg(S)] \)

The Girsanov transform can be used to simplify expressions of the form \( E^Q[Sg(S)] \). If \( S(T) \) is an \( N \)-dimensional log-normal price process with correlation \( \{D\}_{ij} = Corr[dW_i, dW_j] \) under probability measure \( Q \). Then

\[
E^Q[S_{m}(T)g(S(T))|\mathcal{F}_0] = S_{m,0} e^{rT}E^Q[g((\xi_{m,1} S_1(T), \ldots, \xi_{m,N} S_N(T))|\mathcal{F}_0], \quad (8)
\]

where \( \xi_{mi} = \exp(\sigma_i \sigma_m \{D\}_{im} = \exp(\frac{1}{2\pi} Cov[\log dS_i(T), \log dS_m(T)]) \). Details and a proof can be found in Rasmusson’s dissertation[18].

2.4 Definitions

The network is modelled as an undirected weighted graph. The connections are edges in the graph and the exchange and access points are vertices. The
weights are network capacity. Routes from one point in the network to another are paths in the graph. A set of paths in the network can be described as a network capacity matrix, \( V(t) \) with elements \( \{v_{im}\} \). Each edge \( m \) is given a capacity weight \( v_{im} \) for each path \( i \). In figure 4 an example of a network with seven connections is given. If capacity \( c_A \) is sent along path \( A = \{1, 2, 7\} \) and capacity \( c_B \) is sent along path \( B = \{1, 4, 5, 7\} \), the capacity matrix is

\[
V = \begin{bmatrix}
c_A & c_A & 0 & 0 & 0 & 0 & c_A \\
c_B & 0 & c_B & 0 & c_B & 0 & c_B \\
\end{bmatrix}.
\] (9)

2.4.1 The cost of resources

The capacity prices in a network with \( N \) connections are \( \{S_m(t)\}_{m=1}^N \). This can be written as an asset price vector \( S(t) \). Let \( S_m(t) \) be the price of capacity on router \( m \) at time \( t \), \( S_m(0) = S_{0,m} \). The cost of resources along path \( i \) in the network is

\[
C_i(t) = \sum_{m=1}^N C_{im}(t) \\
= \sum_{m=1}^N v_{im}S_m(t)
\] (10)

where \( N \) is the number of connections in the network and \( v_{im} \) is the amount of capacity needed. The set of \( M \) cost of resources can be written as a vector \( C(t) = V(t)S(t) \) with elements \( \{C_i(t)\}_{i=1}^M \):

\[
\begin{bmatrix}
C_1(t) \\
\vdots \\
C_M(t)
\end{bmatrix} = \begin{bmatrix}
v_{i1} & \cdots & v_{iN} \\
\vdots & \ddots & \vdots \\
v_{M1} & \cdots & v_{MN}
\end{bmatrix} \begin{bmatrix}
S_1(t) \\
\vdots \\
S_N(t)
\end{bmatrix}
\]

2.4.2 The send fee

Even though users have to buy and sell resource shares in order to send traffic some other payment is necessary to price network derivatives. Rasmusson[16]
shows that the price of a future to buy resources on the cheapest path between two network nodes at $T_1$ that are resold at $T_2$ is zero. The buying and selling of resources that are a part of the derivative payoffs will always amount to a bundle future or a sum of bundle futures, since resource holders can be assumed to want to sell the resources when they are done sending.

In order to give resource holders an incentive to release resources Rasmusson uses a send fee. The owner of a resource is allowed to send an amount $v$ of traffic over connection $m$ for a short amount of time if he pays $\varepsilon v S_m(t)\Delta t$, where $\varepsilon \in \mathbb{R}$. If the user wants to send for a longer duration he should pay the send fee at every instant while sending. This may be hard to do in practise, so instead he may pay the discounted expected value of the total send fee at $T_1$. Using this relation from on page 28 in the appendix

$$E^Q \left[ \int_{T_1}^{T_2} S_m(t) dt \right] = S_m(T_1) \frac{(e^{r(T_2-T_1)} - 1)}{r},$$

we get

$$e^{-r(T_2-T_1)}E^Q \left[ \int_{T_1}^{T_2} S_m(t) dt \right] = S_m(T_1) \frac{(1 - e^{-r(T_2-T_1)})}{r},$$

which approaches $(T_2 - T_1)S_m(T_1)$ as $r \to 0$. The price of sending perpetually starting at $T_1$ is $S_m(T_1)/r$.

**The path send fee** The send fee for sending traffic along path $i$ for a short duration $\Delta t$ is

$$\Delta \Pi_i(t) = C_i(t)\Delta t$$

$$= \sum_{i=1}^{N} v_{im} S_m(t)\Delta t$$

We define the path send fee vector as $\Delta \Pi(t) = V(t)S(t)\Delta t$. $\Delta \Pi$ has elements $(\Delta \Pi_i)_{i=1}^{M}$. We also define the total path send fee at time $t$ as

$$\Pi_i(t, t + \tau) = e^{-r\tau}E^Q \left[ \int_{t}^{t+\tau} d\Pi_i \left| F_t \right. \right]$$

$$= \frac{(1 - e^{-r\tau})}{r} \sum_{m=1}^{N} v_{im} S_m(t)$$

for which $(1 - e^{-r\tau})/r \to \tau$ as $r \to 0$. The total path send fee is what a user will have to pay at $t$ to send traffic along path $i$ for a duration $\tau$. 13
Send fee example  To illustrate why a send fee of some sort is necessary, here is an example based on one of Rasmusson’s theorems[16]. Consider the price of sending traffic over connection $m$ between times $T_1$ and $T_2$. At $T_1$ the price $m\times S_m(T_1)$ is paid to get the needed capacity. The send fee of sending traffic with this capacity

$$\int_{T_1}^{T_2} v_m\epsilon S_m(t)dt$$

is paid continuously between times $T_1$ and $T_2$. At $T_2$ the resource is sold for $S_m(T_2)$.

A resource holder who wishes to avoid risk would be interested in buying a contract today, a time $t = 0$ that gives him the cash flows in the previous paragraph. The risk neutral pricing formula gives us a price of this contract:

$$\Pi_0 = e^{-rT_1}E[ v_mS(T_1) | \mathcal{F}_0 ] + e^{-rT_2}E\left[ \int_{T_1}^{T_2} v_m\epsilon S_m(t)dt \middle| \mathcal{F}_0 \right] - v_mS_0$$

$$= e^{-rT_2}E[ v_mS(T_2) | \mathcal{F}_0 ]$$

$$= v_mS_0 + e^{-rT_2}E\left[ \int_{T_1}^{T_2} v_m\epsilon S_m(t)dt \middle| \mathcal{F}_0 \right] - v_mS_0$$

$$= e^{-rT_2}E\left[ \int_{T_1}^{T_2} v_m\epsilon S_m(t)dt \middle| \mathcal{F}_0 \right] - v_mS_0$$

$$= e^{-rT_2}E\left[ \frac{(e^{r(T_2-T_1)} - 1)}{r} S_m(T_1) \right]$$

$$= e^{-rT_2}v_m\epsilon \left( e^{r(T_2-T_1)} - 1 \right)$$

$$= v_m\epsilon \frac{(e^{-rT_1} - e^{-rT_2})}{r} e^{T_1} S_m,0$$

$$= v_m\epsilon S_m,0 $$

Thus, if $\epsilon = 0$, that is, if there were no send fee, the price of the contract would be zero. From here on $\epsilon$ is assumed to be 1.

3 New Internet applications

The Internet2 QoS Working Group is currently doing a survey on QoS needs for new Internet applications[12]. I will only focus on a few applications since a complete survey of the services required for all the new applications is out of scope of this thesis. Here, however, is a brief survey of applications based on the classes of applications discussed in the Internet2 survey, and their QoS requirements.

3.1 Auditory applications

Applications related to sound can be divided into interactive and non-interactive auditory applications.
**Interactive** The most common form of communication on earth is voice communication. *Conversational audio* is an interactive auditory application that enables voice communication over long distances. The interactivity of these applications requires some level of QoS, but not as high as some of the more advanced applications.

**Non-interactive** Professional quality audio streaming applications will be used to distribute music. It has to be high-sampling, multichannel audio with CD-equivalent or better quality. The streams may need to be transmitted uncompressed or losslessly compressed to maintain quality. For this application delays in the order of seconds are acceptable. When demands on timing are higher, such as when sending a live concert, *high quality audio orchestration* applications are used. They require higher levels of QoS since end-to-end delay, jitter and packet-loss are crucial factors.

### 3.2 Video-based applications

Just like auditory applications, video-based applications come in interactive and non-interactive varieties. Interactive applications are generally more sensitive to delays than non-interactive applications. On the other hand users will probably tolerate certain video distortion in long-distance collaboration, so requirements on packet loss and bandwidth are less stringent.

**Interactive** The quality requirements of video-based applications differ for interactive and non-interactive applications. High quality audiovisual conferencing, or simply video-conferencing, is an interactive video-based application. The quality requirements are dependent on the exact nature of the conferencing application. To achieve a collaborative conference experience this may involve orchestrating both audio and video, which requires a high level of QoS.

**Non-interactive** Examples of non-interactive video-based applications are *video streaming* and *high definition TV*. Video streaming can either be

- Real-time dissemination of live events, such as news or sports. This involves transmitting video to a large group of users. These applications are best serviced in a scalable fashion by multicast networks. (For more on the quality requirements of multicasting, see section 4.1 on page 17.)

- Streaming of on-demand pre-recorded, stored video material from a remote server. The latency demands may be slightly relaxed which means that low levels of jitter can be alleviated with appropriate buffering algorithms and lost packets can retransmitted. This in turn relaxes the network quality requirements.

The objective of High definition TV, or HDTV, is to provide high-resolution, high-quality moving images at qualities comparable or better than any contemporary digital equivalent such as DVD, by far surpassing today’s TV experience. HDTV comes in different varieties for different target users, so the requirements also vary. Bandwidth requirements range from 19.2Mbps for consumer
grade quality (broadcast quality MPEG-2) to 1.5Gbps for raw, or uncompressed, HDTV used by studios in some stages of production. If nothing else, HDTV requires a lot of bandwidth and multicasting seems to be the only solution.

3.3 Remote control of instruments

To be able to control instruments over a great distance is useful in many areas of science, especially medicine where it can be used in e.g. robotic surgery. Quality requirements vary with applications.

**Robotic surgery** The main reason for using robots in medicine is to reduce the people needed to operate. Today nearly a dozen people are needed to perform an operation. In the future, surgery may require only one surgeon, an anesthesiologist and one or two nurses. The surgeon performs the operation at a console. This console could be in the operating room or, if the network quality is sufficient, at a location a long distance from the patient[4]. The quality requirements are high. High resolution images and/or hi-fi video feeds will be transmitted to the user. This will require quality similar to that of interactive video. The remote control data sent to the remote instruments also has high quality requirements, but of a different kind. The bandwidth is relatively low, but the requirements on quality measurements related to reliability, e.g. delay variation and packet loss, are high.

Other examples are remote control of large telescopes (e.g. the SOAR telescope) and remote access of powerful microscopes (e.g. the MAGIC center at Cal State Hayward)[12].

3.4 Grid computing

Grid computing is a way for computational scientists to share both computational resources and data over a network. Many complex systems demand computational, storage and networking resources on a very large scale. Current grid computing projects are e.g. the EU sponsored DataGrid Project, GriPhyN – Grid Physics Network, funded by NSF and NEESgrid for the earthquake engineering community, also funded by NSF. Grid systems involves many kinds of applications with different quality requirements.

The European DataGrid Project is intended to enable three virtual organizations: the High Energy Physics community led by CERN in Switzerland, the Biology and Medical Image processing community led by CNRS in France and the Earth Observations community led by the ESA/ESRIN in Italy. A Virtual Organization is a distributed community of institutions and individuals willing to share their resources in order to achieve common goals. The idea is that community members should have access to powerful resources without having to know where the resources come from. Users simply submit requests to the Grid specifying application-level requirements and provides input data. The Grid then finds and allocates suitable resources to satisfy these requests. The processing is monitored by the Grid, and the user is notified when the results are ready to be presented[15].
3.5 Virtual Environments

A virtual environment is a computer generated simulation, often with an interface that uses 3D computer graphics. There are many approaches to developing applications that utilize computer generated 3D worlds. Here are a few examples of concepts related to research on virtual environments. The network capacity requirements are similar to those of interactive visual applications.

DVE’s A Distributed Virtual Environment, or DVE, is a virtual environment that is multi-user and distributed – i.e. a virtual environment supporting multiple interacting users and running on several computers connected by a network[7]. DIVE, developed by SICS in Kista, is an example of a platform for DVE’s. The DIVE platform uses multicast and partitioning of the virtual universe to allow a large number of simultaneous participants[5].

Mixed Reality Mixed Reality is an interface that overlays digital images onto those of the real world. 3-D objects are merged into the real environment placing graphical information directly in the viewpoint of the user. The National University of Singapore has done research in this area and claim that the technology will be available in one to two years[11].

Tele-immersion Tele-immersion creates the illusion of two people, possibly on opposite sides of the globe, being in the same physical space as each other. It enables users at geographically distributed sites to collaborate in real time in a simulated environment as if they were in the same room. The ultimate goal of tele-immersive research is to construct a holodeck similar to the one seen on the Star Trek TV-series. E.g. the National Tele-immersion Initiative - NTII did research on this until recently[13].

4 Network services

In this chapter five services will be described and priced as financial derivatives. As mentioned earlier (2.4.2) the cost of resources (equation 10) part of the derivative contracts will always amount to a bundle future, which costs nothing. This is why the send fee is the only part of the cost mentioned when pricing services in this chapter.

4.1 Reliable multicast

Service The user gets the capacity needed to multicast data for some duration.

What multicast is Multicasting is a bandwidth conserving technology that reduces traffic by simultaneously delivering a single stream of information to thousands of recipients. This is possible today, but multicasting requires high service quality in order to be useful for all applications. Since network capacity
is shared by many users when multicasting data, the capacity costs are lower than it of unicasting the same information.

The conventional way of networking, sometimes called unicast, is by sending data from one computer to another. This is inefficient when there are many recipients of the same information since separate identical data streams has to be generated for each recipient. A copy of the data is sent to each client, regardless of whether the paths to clients are similar.

When multicasting data the sending is done in such a way that the data is sent in just one copy along any part of the network. A copy is made only when the paths to two recipients differ. The most cost-effective way of doing this is by constructing a tree that spans every member and minimizes the costs of sending. Such a tree is called an optimal spanning tree. Packets are replicated by routers at the tree nodes and sent to a 'host-group' consisting of zero or more hosts. The participants of the host-group has to be listed as members in order to receive data.

This service is intended for wide-area multicast transmission on transit networks. Transit networks are networks that transfers traffic to other networks in addition to carrying traffic for their own hosts. Wide-area multicast is implemented on the Network layer (see 2.1 on page 7). Local-area multicast does not require the use of multicast routers, but instead uses the multicast transmission capabilities of the Physical layer, in other words, works on a lower level than wide-area multicast.

In addition, the service only guarantees capacity on the Internet-wide part of the wide-area network. At the leaves of the spanning tree multicast routers use a protocol such as IGMP to communicate with receivers on leaf networks[21].

**Problems with multicast**  In order to be used for transmission of e.g. video, multicasting has to be more reliable than it is today. Multicasting is a push technology – the server sends data to the client without the client requesting it. Clients cannot request retransmissions of lost data, since this would mean that other members of the host-group had to wait for a retransmission that they didn’t need. Receiving applications cannot adapt to packet loss by requesting lost data. This is a problem for applications that are sensitive to lost data.

Spamming is another problem. Today’s email system, also a push technology, suffers from spamming. Data sent by email must however be copied, whereas multicast data need only be sent in one copy. Thus, multicasting could be used to spam millions of receivers with large amounts of data they have not requested. The send fee and cost of resources does not provide the sender with an incentive to avoid unnecessary multicasting, since the sender can send data in the best effort class for free. The problem could be solved e.g. by introducing a new multicast fee, or by requiring most of the multicast data to be sent in the first-class class.

**Applications**  Multicasting can be used whenever data is sent from one user to many users – one-to-many. Non-interactive video-based transmissions with many receivers (in other words, TV) is the obvious application and perhaps also the most important one.

Multicasting can also be used in a number of applications such as video con-
ferencing, virtual environments and tele-immersion, when several users want to both send data to and receive data from many users – a many-to-many situation. We will focus on services for situations when data is sent from one sender to many recipients.

**Payment and delivery** The user pays for the contract at time 0 and the capacity is delivered at $T$. At $T$ the user gets enough capacity to multicast traffic with capacity $v$ for a duration $\tau$ along the path in the tree that connects the user with the sender. Figure 5 shows three users connected to access points at the leaves of a multicast spanning tree. Although the capacity is $v$ for each connection in the tree, the capacity needed for one of the recipients of a one-to-many transmission varies along the path. The first part of the path is shared among all the members of the multicast group. At each node of the tree, each user’s share of the capacity increases, until a leaf of the tree is reached. At each leaf the capacity is possibly still shared by a large number of users.

**Pricing** The price is, as usual, the discounted expected value of the capacity needed. The path of one user is not necessarily the minimum cost path. It will probably be beneficial to all users in the host-group if the paths are chosen so that the total cost of all the paths in the spanning tree is minimized. This means that we need to know something about the expected spanning tree in order to price the service. The structure of the spanning tree is dependent on the prices of capacity and the other users of the multicast service.

Here’s an example of an approach to multicast service pricing. Let the required capacity be constant and the same for all users and paths: $v$. The capacity
matrix is defined as on page 12 and describes the paths of one user. Figure 6 on the preceding page shows one of the paths, path $i$ of one of the users. The path consists of connections $1$ to $n_i = n$. The send fee for multicasting data for a very short duration, $\Delta t$ starting now, at $t = 0$, along this path, assuming that this path is part of the multicast spanning tree, is

$$\sum_{k=1}^{n_i} \frac{v S_k(t)}{N_{ik}(t)} \Delta t,$$

where $N_{ik}$ is the number of users sharing connection $k$ on path $i$ if the path is part of the multicast spanning tree. The send fee of sending for a longer duration is

$$\sum_{k=1}^{n_i} \int_0^{\tau} \frac{v S_k(t)}{N_{ik}(t)} \, dt,$$

The variables $S_k(t)$ and $N_{ik}(t)$ are stochastic for $t > 0$. In addition, we do not know which of the paths will be part of the multicast spanning tree. This uncertainty can be represented by a switch function:

$$1_{\{M_i\}},$$

which is 1 if $M_i$, the event that path $i$ is part of the multicast spanning tree, occurs and 0 if not. The price of a multicast forward contract with delivery date $T$ can then be written as

$$v \cdot e^{-r(T+\tau)} \mathbb{E}^Q \left[ \sum_{k=1}^{n_i} \int_T^{T+\tau} \frac{S_k(t)}{N_{ik}(t)} \, dt \cdot 1_{\{M_i\}} \right]_{F_0}, \quad (14)$$

a formula that may be hard to evaluate. The above is one of many approaches to pricing multicast services. For example, the switch function, $1_{\{M_i\}}$ and the variable $N_{ik}(t)$ in the equation above are both dependent on the structure of the tree, including the number of users at each leaf. They could be substituted for one variable that has a different value of the form $1/N$ for each possible multicast spanning tree. The problem with this is that the number of possible spanning trees for a network with millions of users could be huge, much larger than the number of possible paths to one user.

It may be necessary to simplify the problem in order to find a feasible pricing system. An example of that is the Backbone pricing that follows, were we confine ourselves to a small but important part of the tree.

**Backbone pricing** The exact structure of the future multicast spanning tree is unknown. However, the structure of the first main branches of the tree, which is part of a backbone network will be easy to predict or even specified in advance. Pricing multicast contracts on the backbone tree is easier than the multicast pricing problem stated above, if the structure of the tree is known.

Let’s look at the price of multicasting along one of these connections, connection $k$, for one user. With similar notation as above, the price is
\[
v \cdot e^{-r(T+\tau)} E_Q \left[ \int_T^{T+\tau} \frac{S_k(t)}{N_k(t)} \, dt \bigg| \mathcal{F}_0 \right]
\]

where \( N_k \) is the total number of users on the connection. We still need to know the distribution of the number of users on each connection in order to evaluate the price of a multicast service.

4.2 Video on-demand

Service The user gets the capacity needed to send data from one of a number of servers for a specified duration. The user chooses the time of delivery, but the server and path from server to user \( i \) chosen by the broker.

Applications This is a service for situations when the user wants a certain data stream, but doesn’t care about where it comes from or which way it takes to get to him. Video on-demand is one such application. As mentioned earlier, delays may be acceptable when streaming video, but video require a lot of bandwidth, so it may be necessary to reserve capacity by buying a service.

![Diagram](image)

Figure 7: When the user exercises his contract, the broker selects servers and paths and delivers the capacity.

Payment and delivery The user pays for the contract at time 0 and may exercise it at any time between 0 and \( T_1 \). When the contract is exercised the user gets capacity to receive traffic sent along the cheapest path for a duration \( \tau \).

Pricing There is at least one possible path for each server, but there can also be more than one path to choose from from each server. The paths and the needed capacity are both known when the contract is written. At the time of exercise the user receives the send fee for sending with a specified capacity for some duration \( \tau \) along the least cost path. Let the interval be \([0, T_1]\), the strike price \( K \) and the number of paths be \( L \). Even though most users of the service will not choose the time of delivery so as to maximize their profit, this is what
we have to assume when pricing the contract. The contract is an American call option with the price

\[ e^{-rT_1} \mathbb{E}^Q \left[ \max_{T \in [0,T_1]} \left( \max \left( \frac{1}{r} \sum_{m=1}^{N} v_i m \int_{T}^{T+\tau} S_m(t) dt \right) - K, 0 \right) \right] \mid \mathcal{F}_0 \]

where \( T \) is the actual time of exercise and \( T_1 \) is the latest time of exercise. By using equation 25 this can be rewritten as

\[ e^{-rT_1} \frac{(e^{r\tau} - 1)}{r} \mathbb{E}^Q \left[ \max_{T \in [0,T_1]} \left( \max \left( \frac{1}{r} \sum_{m=1}^{N} v_i m S_m(T) \right) - K, 0 \right) \right] \mid \mathcal{F}_0 \]

for which \( \lim_{r \to 0} e^{-rT_1} \frac{(e^{r\tau} - 1)}{r} = \tau \). This is similar to an American basket option. The only difference between this derivative and Rasmusson’s network option[19] is that this service allows the user to choose the time of delivery. Pricing and hedging multi-asset basket options with both high dimensionality and early exercise is a hard problem. A numerical algorithm for pricing American basket options has been suggested by Wan[23], but this option is more complicated. The price of this service may possibly be evaluated by using Rasmusson’s method of evaluating the PDF[17] or by using Edgeworth expansion[9] (section B.1 in the appendix) to approximate the distribution of the asset basket.

4.3 Internet Capacity Boost

Service The user gets an Internet Capacity Boost that raises the perceived level of quality when using different Internet applications. The intention is to give a lot of freedom to the user, but the choices that the user is allowed to make make the service more expensive.

Applications This is an attempt to make a service that can be used for many different purposes. The contract can be exercised any time the user perceives low quality while using a number of different applications. The user should think of the service as an Internet Capacity Boost that increases the throughput and reduces delays. In figure 8 the user has three sets to choose from. If he is sending data from either London or New York he chooses one of those sets. If the user is sending video he chooses the server set, which exercises a video on-demand contract like the one described above.

Payment and delivery The user chooses the time of exercise and a set of paths. The capacity is specified in advance. The broker then chooses one of the paths in the set.

Pricing The pricing is similar to the pricing of the Video on-demand service above. The difference is the choice of paths the user is allowed to make. We have to assume that the user selects the most expensive set of paths. This means that we loose some of the effect of allowing the least cost path be chosen at the
time of delivery. The two effects will offset each other – a choice made by the user makes the service more expensive, a choice by the service provider makes the service cheaper. A pricing formula, with $M$ path sets each with a subset of $L_k$ paths, $k = 1 \ldots M$:

$$e^{-rT_1} \left( \frac{e^{r\tau} - 1}{r} \right) \mathbb{E} \left[ \max_{T \in [0, T_1], k=1 \ldots M} \left( \max_{i=1}^{L_k} \left( \min_{m=1}^{N} v_{im} s_m(T) \right) - K, 0 \right) \right] \mathcal{F}_0$$

for which $\lim_{r \to 0} e^{-rT_1} (e^{r\tau} - 1)/r = \tau$. This service may possibly be evaluated using one of the methods suggested in section 4.2, but the many min and max functions in the formula may make pricing too time-consuming.

### 4.4 Time-dependent capacity

**Service** A user may need different quality levels at different times. This service gives the user the right to send traffic along a specified path for a specified duration of time, just like the total path fee in section. The difference is that the capacity needed is allowed to be different at different times, even during sending.

**Applications** Many users are probably going to need different quality levels at different times of day or on different days of the week. This service can be used as a building block to make new services with time-dependent capacity.

**Payment and delivery** The user pays for the service at the same time that the capacity rights are delivered. The user gets the right to send capacity for a duration $\tau$ along path $i$. The capacity matrix $V$ is a predetermined function of time, $V(t)$ with elements $\{v_{ij}(t)\}$. The function could for example be periodical with a period of a day, week, month or year.
**Pricing**  The cost of resources (equation 10 on page 12) becomes

\[ C_i(t) = \sum_{m=1}^{N} v_{im}(t) S_m(t) \]  \hspace{1cm} (15)

The path send fee (equation 12) becomes

\[ \Delta \Pi_i(t) = \sum_{m=1}^{N} v_{im}(t) S_m(t) \Delta t \]  \hspace{1cm} (16)

and the total path send fee at \( t \), when traffic is sent from \( t \) for at duration \( \tau \)

\[ \Pi_i(t, t+\tau) = e^{-r\tau} \mathbb{E}^Q \left[ \int_t^{t+\tau} d\Pi_i \left| F_t \right. \right] 
= e^{-r\tau} \mathbb{E}^Q \left[ \int_t^{t+\tau} \sum_{m=1}^{N} v_{im}(s) S_m(s) ds \left| F_t \right. \right] 
= e^{-r\tau} \sum_{m=1}^{N} \mathbb{E}^Q \left[ \int_t^{t+\tau} v_{im}(s) S_m(s) ds \left| F_t \right. \right] 
= e^{-r(t+\tau)} \sum_{m=1}^{N} S_m(t) \int_t^{t+\tau} v_{im}(s)e^{rs} ds 
= \sum_{m=1}^{N} S_m(t) \int_t^{t+\tau} v_{im}(s)e^{r(s-t-\tau)} ds \]  \hspace{1cm} (17)

which approaches \( \sum_{m=1}^{N} S_m(t) \int_t^{t+\tau} v_{im}(s) ds \) as \( r \to 0 \). Equation 18 reduces to \( \frac{(1-e^{-r\tau})}{r} \sum_{m=1}^{N} v_{im} S_m(t) \) from equation 13 on page 13 when the capacity is constant, \( v_{im}(t) = v_{im} \). Time-dependent capacity does not seem to complicate the pricing of services based on a send-fee, since the total path send fee can be written on the form \( \sum_{m=1}^{N} S_m(t) f_{im}(t, \tau) \).

### 4.5 Traffic sometime during the night

**Service**  The user gets the right to send traffic at requested capacity for a requested duration at some time during a period such as a night.

**Applications**  The service is intended for situations when the exact time of transmission is not important, but the capacity is. The idea is to lower the price of the service by letting the provider choose the time of delivery. As it turns out, this does not make the service cheaper, due to the Martingale properties of the total path send fee.

**Payment and delivery**  The user pays for the service at time 0. The path, duration and capacity are specified. The broker chooses the time of delivery.
Pricing  The exact time of transmission is not chosen by the buyer, but by the seller of the contract, the broker. The broker chooses the time so as to minimize the cost. The pricing should be based on the assumption that the seller chooses this time in an optimal way. To price the service we need to know when the broker can be expected to exercise the contract. One approach is dynamic programming. Dynamic programming solves a problem step by step, starting at the termination time and working back to the beginning[6]. We will use the dynamic programming approach to see if the service provider can reduce his expected cost by choosing the time of delivery in an optimal way.

Let the capacity be \( v \) for all connections, the duration be \( \tau \) and the interval be \([T_1, T_2]\). We also assume that the path is specified. At every instant between times \( T_1 \) and \( T_2 - \tau \) the seller has two options – to either deliver the capacity or to wait.

**Step 1.** At \( T_2 - \tau \) he has no choice but to deliver the capacity in order to fulfill his obligation. The total path send fee (equation 13 on page 13) that would be paid at \( T_2 - \tau \) for sending along some path \( \{1, 2 \ldots N\} \) for the duration \( \tau \) is
\[ \Pi_{T_2 - \tau, T_2} = \mathbb{E}^Q \left[ \int_{T_2 - \tau}^{T_2} d\Pi_i \left| \mathcal{F}_{T_2 - \tau} \right] \right] = \frac{(1 - e^{-r\tau})}{r} \sum_{m=1}^{N} v_{im} S_m(T_2 - \tau) \]

**Step 2.** Earlier, at \( T_2 - \tau - \Delta t \), the seller still has a choice of either delivering the capacity, which means paying the total path send fee

\[
\Pi_{T_2 - \tau - \Delta t, T_2 - \Delta t} = e^{-r\Delta t} \mathbb{E}^Q \left[ \int_{T_2 - \tau - \Delta t}^{T_2 - \Delta t} d\Pi_i \left| \mathcal{F}_{T_2 - \tau - \Delta t} \right] \right] = \frac{(1 - e^{-r\tau})}{r} \sum_{m=1}^{N} v_{im} S_m(T_2 - \tau - \Delta t) \tag{19}
\]

or waiting, which should be seen as costing the discounted expected value of the total path send fee of equation 19:

\[
e^{-r\Delta t} \mathbb{E}^Q \left[ \left(1 - e^{-r\tau} \right) \sum_{m=1}^{N} v_{im} S_m(T_2 - \tau) \right] = \frac{(1 - e^{-r\tau})}{r} \sum_{m=1}^{N} v_{im} S_m(T_1 - \tau - \Delta t)
\]

\[
e^{-r\Delta t} \mathbb{E}^Q \left[ \left(1 - e^{-r\tau} \right) \sum_{m=1}^{N} v_{im} S_m(T_1 - \tau - \Delta t) \right]
\]

The expected cost is the same whether he decides to deliver or not at \( T_2 - \tau - \Delta t \). With the right choice of \( \Delta t \) this could be any time between \( T_1 \) and \( T_2 - \tau \). We conclude that the provider of the service cannot use the choice to lower his expected cost. This is because the discounted total path send fee has the martingale property, that is,

\[
\mathbb{E}^Q \left[ e^{-r(T_2 - T_1)} \Pi_{T_2, T_2 + \tau} \right| \mathcal{F}_{T_1} ] = \Pi_{T_1, T_1 + \tau} \tag{21}
\]

for any \( T_2 \geq T_1 \). We may assume that the broker chooses the time arbitrarily. The price of the contract at \( t = 0 \), assuming that the contract is delivered at some arbitrary point in time \( t \), is

\[
e^{-rt} \mathbb{E}^Q \left[ \left(1 - e^{-r\tau} \right) \sum_{m=1}^{N} v_{im} S_m(t) \right| \mathcal{F}_0 \right] = \frac{(1 - e^{-r\tau})}{r} \sum_{m=1}^{N} v_{im} S_{m,0} \tag{22}
\]

Note that the price is independent of the time of delivery. The pricing equation approaches \( \tau \sum_{m=1}^{N} v_{im} S_{m,0} \) as \( r \to 0 \). The price is easy to evaluate since it is a linear function of the current capacity spot prices along the path.
5 Discussion

We have used Rasmusson’s model to suggest and discuss the pricing of five network capacity services. The Black-Scholes model was used to model the services as financial derivatives of network capacity resources. A send fee as suggested by Rasmusson et al.[19] has been used when pricing the derivatives. The send fee gives users an incentive to sell resources when they are done sending. The services suggested and priced in this thesis tend to involve the sum of assets called the cost of capacity, a weighted sum of log-normal variables. This indicates that Rasmusson’s method[17] for evaluating the PDF for such sums should be very useful for pricing network services. Rasmusson’s method does not rely on the assumption that the weighted sum has a log-normal distribution. An alternative method that does rely on such an assumption is approximation of the PDF by Edgeworth expansion[9].

In section 4.1 a service for multicasting was suggested. The future structure of the tree and the future number of users on each connection in the tree are both uncertain. We need to know something about the distribution of variables that describe the two uncertainties. A users decision whether to buy a multicast service, is not only based on the network prices, but also possibly based on exogenous factors such as prices on some other market. To price multicast services we need a model that include assumptions about the decisions of the multicast users.

In some of the services suggested here the user and/or the broker has been allowed to make choices. It is a well know feature of the Black-Scholes model that buyers generally have to pay more for choices. In the services of section 4.2 and 4.3 the user was allowed to choose the time of delivery and the broker was allowed to chose the path from a set of paths. The choice of time by the user raises the price (with an amount ≥ 0) and the choice of paths lowers the price by the user (with an amount ≥ 0). These and other services illustrate how giving choices to either the user or the broker can either raise or lower the price of services.

The “Traffic sometime during the night” service of section 4.5 turned out to be an example of a situation when giving a choice to the broker does not make the service cheaper, at least not when using this model. The expected cost of delivering was the same whenever he decided to deliver the capacity, so he could not use the choice to lower his cost. This is because the discounted total path send fee has the martingale property (equation 21 on the page before).

Allowing the requested capacity to change deterministically over time does not seem to make the pricing of services more difficult. The objective of the service in section 4.4 was to see if a total path send fee as defined by equation 13 could be priced when the capacity was time dependent. The resulting equation indicates that time-dependent capacity is not a problem when pricing services using a send fee.

A Network derivatives

Cheapest path price The price at $T_1$, of sending traffic along the cheapest path between times $T_1$ and $T_2$ is
A bandwidth user may want to know in advance how much he will have to pay at time 0 if event A occurs at T1, 0 if not.

\[ \mathbb{E}^Q \left[ \int_{T_1}^{T_2} S_m(t) dt \bigg| \mathcal{F}_{T_1} \right] = \mathbb{E}^Q \left[ \int_{T_1}^{T_2} S_{m,0} \exp \left\{ (r - \sigma^2/2)t + \sigma W_t \right\} dt \bigg| \mathcal{F}_{T_1} \right] \]

= \mathbb{E}^Q \left[ \lim_{\Delta t \to 0} \sum_{t_i=T_1/\Delta t}^{T_2/\Delta t} S_{m,0} \exp \left\{ (r - \sigma^2/2)t_i + \sigma W_{t_i} \Delta t \right\} \Delta t \bigg| \mathcal{F}_{T_1} \right]

= \lim_{\Delta t \to 0} \sum_{t_i=T_1/\Delta t}^{T_2/\Delta t} S_{m,0} \exp \left\{ (r - \sigma^2/2)t_i \Delta t \right\} \Delta t \exp \left\{ \sigma W_{T_1} + \sigma^2/2(t_i \Delta t - T_1) \right\}

= S_{m,0} \exp \left\{ \sigma W_{T_1} - \frac{\sigma^2}{2} T_1 \right\} \int_{T_1}^{T_2} e^{rt} dt

= S_m(T_1) \left( e^{r(T_2 - T_1)} - 1 \right). \quad (25)

which approaches \((T_2 - T_1)S_m(T_1)\) as \(r \to 0\). \(1_A\) is 1 if event A occurs at \(T_1\), 0 if not.

**A.1 Pricing example - Network forward**

A bandwidth user may want to know in advance how much he will have to pay at \(T_1\) for the right to send traffic between times \(T_1\) and \(T_2\). The forward price is the expected value under the risk-neutral measure of how much he has to pay at \(T_1\). If the forward price is discounted with \(e^{-rT_1}\) we get the price that the user would have to pay today \((t = 0)\) to get right to send traffic between times \(T_1\) and \(T_2\).

Since the buying and selling of resource shares will amount to a bundle future, the forward price at time 0 is the expected value of the total send fee:
\[ \Lambda = \mathbb{E}^Q [\Pi_{T_1, T_2} \mid \mathcal{F}_0] \]
\[ = \mathbb{E}^Q \left[ \frac{1 - e^{-r(T_2 - T_1)}}{r} \sum_{m=1}^{M} \sum_{i=1}^{N} v_{im} S_m(T_1) \cdot 1_{\{C_i = \min_k C_k\}} \mid \mathcal{F}_0 \right] \]
\[ = \frac{1 - e^{-r(T_2 - T_1)}}{r} \sum_{m=1}^{M} \sum_{i=1}^{N} v_{im} \mathbb{E}^Q [S_m(T_1) \cdot 1_{\{C_i = \min_k C_k\}} \mid \mathcal{F}_0] \]
\[ = \frac{1 - e^{-r(T_2 - T_1)}}{r} \sum_{m=1}^{M} \sum_{i=1}^{N} v_{im} e^{rT_1} S_{m,0} \mathbb{E}^Q \left[ 1_{\{\hat{C}_{im} = \min_k \hat{C}_{km}\}} \right] \mathcal{F}_0 \]
\[ = e^{rT_1} \frac{1 - e^{-r(T_2 - T_1)}}{r} S_{m,0} \sum_{m=1}^{M} \sum_{i=1}^{N} v_{im} \mathbb{Q} \left( \hat{C}_{im} = \min_k \hat{C}_{km} \mid \mathcal{F}_0 \right). \quad (26) \]

where \( \hat{C}_{im} = \sum_k v_{ik} \xi_{mi} S_k(T_1) \) and \( \xi_{mi} = \exp \left\{ \frac{1}{2} \text{Cov}^Q[\log dS_i(T), \log dS_m(T)] \right\} \) is the adjusted cost of path \( i \) after a Girsanov transform to eliminate \( S_m(\ldots) \).

For equation 26 we have \( \lim_{r \to 0} e^{rT_1} (1 - e^{-r(T_2 - T_1)})/r = T_2 - T_1 \).

### A.2 Pricing example – Network cash-or-nothing option

The payoff is some predetermined value \( K_{\text{cash}} \):

\[ K_{\text{cash}} \cdot 1_{\{\Pi_{T_1, T_2} < K\}}. \]

The price is, with \( A = \frac{(1 - e^{-r(T_2 - T_1)})}{r} \),

\[ \Gamma = e^{-rT_1} \mathbb{E}^Q \left[ K_{\text{cash}} \cdot 1_{\{\Pi_{T_1, T_2} < K\}} \mid \mathcal{F}_0 \right] \]
\[ = e^{-rT_1} K_{\text{cash}} \mathbb{E}^Q \left[ 1 \left\{ \sum_{i=1}^{M} C_i \cdot 1_{\{C_i = \min_k C_k\} < K} \right\} \mid \mathcal{F}_0 \right] \]
\[ = e^{-rT_1} K_{\text{cash}} \mathbb{E}^Q \left[ 1_{\{\min_i C_i < K\}} \cdot 1_{\{C_i = \min_k C_k\}} \mid \mathcal{F}_0 \right] \]
\[ = e^{-rT_1} K_{\text{cash}} \mathbb{Q} \left[ \min_i C_i < K \mid \mathcal{F}_0 \right] \]
\[ = e^{-rT_1} K_{\text{cash}} \mathbb{Q} \left[ \min C_i < K \mid \mathcal{F}_0 \right] \quad (27) \]

### B Basket options

#### B.1 Valuation of basket options using Edgeworth-series expansion

A basket option is an option whose payoff depends on the value of a portfolio of assets. Hyunh suggested a method of pricing basket options using the Edgeworth-series expansion[9]. A series expansion of the true distribution of
the so-called pseudo basket spot price at maturity is derived. This distribution can then be used to calculate an approximation of the price of a basket option. We define the spot price of a basket of assets as

\[ B(t) = \sum_i a_i S_i(t), \]

where \( S_i(t) \) is the price of asset \( i \) at time \( t \) and \( a_i \) is the quantity of asset \( i \). The distribution of the spot price of the basket at maturity is a convolution of its components’ distributions. The price is a sum of log-normal variables, but the sum of log-normal variables is not log-normal. For a proof that the sum of two log-normal variables is not log-normal, see appendix B.2 on the next page.

We also define the pseudo spot price of asset \( i \) at maturity \( t \) as

\[ \tilde{S}_i(t) = S_i(t) \mathbb{E}^Q[S_i(t)]. \]

The pseudo basket spot price is

\[ \tilde{B}(t) = \sum_i c_i \tilde{S}_i(t), \quad c_i = \frac{a_i \mathbb{E}^Q[S_i(t)]}{\sum_i a_i \mathbb{E}^Q[S_i(t)]}. \]

The value of a European basket call option of maturity \( T \) and strike \( K \) is

\[ V_{C_{Basket}}(T) = \max(B(T) - K, 0) \]

\[ = \sum_i a_i \mathbb{E}^Q[S_i(t)] \max(\tilde{B}(T) - \tilde{K}, 0), \]

where \( \tilde{K} = K/\sum_i a_i \mathbb{E}^Q[S_i(t)] \) is the pseudo strike price. If we approximate the pseudo spot price of the basket \( \tilde{B}(T) \) with a stochastic variable \( \tilde{Z} \) which is log-normal \((\alpha, \beta^2)\), that is \( \tilde{Z} = e^U \) where \( U \in N(\alpha, \beta^2) \), the price is

\[ C_{Basket}(T) = \sum_i a_i \mathbb{E}^Q[S_i(t)] e^{-rT} \int_{-\infty}^{\infty} \max(e^{\alpha + \beta z} - \tilde{K}, 0) \varphi(z) dz \]

\[ = \sum_i a_i \mathbb{E}^Q[S_i(t)] e^{-rT} \left( e^{\alpha + \beta/2} \Phi(d_1) - \tilde{K} \Phi(d_2) \right), \]

where \( r \) is the interest rate, \( d_1 = (\beta^2 + \alpha - \log \tilde{K})/\beta, d_2 = d_1 - \beta, \varphi(z) = \exp(-z^2/2)/\sqrt{2\pi} \) and \( \Phi(z) \) is the distribution function of the standard normal distribution. We can now use the method of moment matching to approximate the price. We then choose \( \alpha \) and \( \beta \) such that \( \mathbb{E}^Q[\tilde{B}(T)] = \mathbb{E}^Q[\tilde{Z}] \) and \( \mathbb{E}^Q[\tilde{B}^2(T)] = \mathbb{E}^Q[\tilde{Z}^2] \). This is a special case of Huynh’s general approach.

Denote by \( F \) the distribution function of \( \tilde{B}(T) \) and by \( f \) its density. \( G \) and \( g \) are assumed to be the distribution function and the density of the approximating log-normal random variable \( \tilde{Z} \). Then, if the moments exist, that is, if \( \mathbb{E}^Q[\tilde{Z}^k(T)] < \infty \) for each \( k = 1, \ldots, n \) it may be shown (see appendix B.3 on page 32) that

\[ f(x) = g(x) + \sum_{k=1}^{n-1} c_k \frac{(-1)^k d^k g}{dx^k}(x) + \varepsilon(x, n), \]

(28)
where \( \varepsilon(x,n) \) is a small error. The constants \( c_k \) can be found by expanding \( \varphi_F(t)/\varphi_G(t) \) as

\[
\frac{\varphi_F(t)}{\varphi_G(t)} = \frac{\mathcal{F}\{f(t)\}}{\mathcal{F}\{g(t)\}} = \sum_{k=0}^{n-1} \frac{c_k (it)^k}{k!} + o(t^n). \tag{29}
\]

where \( \varphi_F(t) \) and \( \varphi_G(t) \) denotes the characteristic functions of \( F \) and \( G \) respectively and \( \mathcal{F}\{f(t)\} \) denotes the Fourier transform of \( f \). The characteristic function \( \varphi_F(t) \) can be written

\[
\varphi_F(t) = \exp \left\{ \sum_{k=0}^{n-1} \kappa_k(F)(it)^k/k! + o(t^n) \right\}
\]

where the constants \( \kappa_k(F) \) are called cumulants of the distribution \( F \). The constants \( c_k \) can be expressed in terms of cumulants \( \kappa_k(F) \) and \( \kappa_k(G) \) which in turn can be expressed in terms of (raw) moments \( \mu_n(F) \) and \( \mu_n(G) \):

\[
\begin{align*}
    c_0 &= 1 & \kappa_0(F) &= 1, \\
    c_1 &= \kappa_1(F) - \kappa_1(G), & \kappa_1(F) &= \mu_1(F), \\
    c_2 &= \kappa_2(F) - \kappa_2(G) + c_1^2, & \kappa_2(F) &= \mu_2(F) - \mu_1^2(F), \\
    c_3 &= \kappa_3(F) - \kappa_3(G) & \kappa_3(F) &= 2\mu_1^3 - \mu_1(F)\mu_2(F) + \mu_3(F), \\
    & \quad + 3c_1(\kappa_2(F) - \kappa_2(G)) + c_1^3 & \kappa_4(F) &= -6\mu_1^4 + 12\mu_1^2(F)\mu_2(F) \\
    & \quad + 3\mu_2^2(F) - 4\mu_1(F)\mu_3(F) + \mu_4(F) & & \vdots \\
    & \vdots \\
\end{align*}
\]

where \( \mu(F) = \mathbb{E}[\bar{Z}] \) is the mean and \( \mu_k(F) = \mathbb{E}[\bar{Z}^k] \) is the k:th central moment. When a number of constants \( c_k \) have been calculated it is possible to compute approximate prices. It can be shown that an approximate price is:

\[
C_{\text{basket}} = \sum_i a_i \mathbb{E}[S_i(t)] e^{-rT} \left( \int_K^\infty (x - \bar{K}) g(x) dx - c_1 \int_K^\infty (x - \bar{K}) \frac{dg}{dx}(x) dx \right. \\
+ \sum_{k=2}^{n-1} \frac{c_k (-1)^k d^k k - 2)g}{dx^k k - 2}(\bar{K}) + \left. \int_K^\infty (x - \bar{K}) \varepsilon(n,x) dx \right).
\]

B.2 Sum of two log-normal variables

We want to show that \( Z = X + Y \), where \( X \) and \( Y \) is independently log-normal(0, 1), does not have a normal distribution.

Consider the stochastic variable \( Z \in \mathbb{R} \) which is log-normal(\( \mu, \sigma^2 \)), that is \( Z = e^U \) where \( U \in N(\mu, \sigma^2) \). The expectation of \( Z \) is

\[
E[Z] = E[e^U] = e^{\mu + \frac{\sigma^2}{2}}. \tag{30}
\]
The moments of $Z$ when $k \geq 1$ are

\[ E[Z^k] = e^{k\mu + \frac{k^2\sigma^2}{2}}. \] (31)

First, second and third moments of $Z = X + Y$ are, since $X$ and $Y$ are independent and have the same distribution:

\[ E[X + Y] = E[X] + E[Y] = 2E[X] = 2e^{\frac{1}{2}} \]
\[ E[(X + Y)^2] = E[X^2] + 2E[X]E[Y] + E[Y^2] = e^2 + \left( e^{\frac{1}{2}} \right)^2 + e^2 = 2e^2 + e \]
\[ E[(X + Y)^3] = 2E[X^3] + 4E[X]E[X^2] = 2e^{\frac{3}{2}} + 4(2e^{\frac{1}{2}} + e^{\frac{3}{2}}). \]

The above and (31) gives us an over determined system of equations:

\[
\begin{align*}
& e^{\mu + \frac{\sigma^2}{2}} = 2e^{\frac{1}{2}} \\
& e^{2\mu + 2\sigma^2} = 2e^2 + e \\
& e^{3\mu + 3\sigma^2} = 2e^{\frac{3}{2}} + 4(2e^{\frac{1}{2}} + e^{\frac{3}{2}}).
\end{align*}
\]

All three equations cannot be satisfied at the same time for any choice of $\mu, \sigma^2$. We have thus shown that

\[ X + Y \notin \text{lognormal}(\mu, \sigma^2). \]

**B.3 Distribution expansion**

We want to show that

\[ f(x) = g(x) + \sum_{k=1}^{n-1} c_k \frac{(-1)^k}{k!} \frac{d^k g}{dx^k}(x) + \varepsilon(x,n), \]

where $G$ and $g$ are assumed to be the distribution function and the density of the approximating log-normal random variable $\tilde{Z}$. $F$ and $f$ are the corresponding functions of $B(T)$.

The expansion (equation 29) of $\varphi_F(t)/\varphi_G(t)$ is

\[ \frac{\varphi_F(t)}{\varphi_G(t)} = \mathcal{F}\{f\}(t) = \mathcal{F}\{g\}(t) = \sum_{k=0}^{n-1} c_k \frac{(it)^k}{k!} + o(t^{n-1}). \]

Multiply this with $\varphi_G(t)$:

\[ \varphi_F(t) = \varphi_G(t) \left( \sum_{k=0}^{n-1} c_k \frac{(it)^k}{k!} + o(t^{n-1}) \right). \]
By using this property of the Fourier transform

\[ (-it)^k \varphi_G(t) = (-it)^k \mathcal{F}\{g\}(t) = (-1)^k (it)^k \mathcal{F}\{g\}(t) = \mathcal{F}\left\{ \frac{d^k g}{dx^k}(x) \right\}(t), \]

we get

\[ \varphi_F(t) = \sum_{k=0}^{n-1} c_k (-1)^k \mathcal{F}\left\{ \frac{d^k g}{dx^k}(x) \right\}(t) + \varphi_G(t) o(t^{n-1}). \]

By doing an inverse Fourier transform we get

\[ f(x) = c_0 g(x) + \sum_{k=1}^{n-1} c_k \frac{(-1)^k}{k!} \frac{d^k g}{dx^k}(x) + \mathcal{F}^{-1}\{\varphi_G(t) o(t^{n-1})\}, \]

which, with \( c_0 = 1 \) and \( \mathcal{F}^{-1}\{\varphi_G(t) o(t^{n-1})\} = \varepsilon(x, n) \), completes the proof.
Index

applications, 14
auditory applications, 14
Aurell, Erik, 5

bandwidth, 6
basket option, 10
spot price of, 30
Black-Scholes
 equation, 10
 model, 9
bundle future, 12
congestion, 5
cost of resources, 12
delay, 6
discussion, 27
DVE, 17
dynamic programming, 25
Edgeworth-series expansion
 valuation using, 29
exchange locations, 7
Feynman-Kac, 10
geometric Brownian motion, 10
 infinitesimal generator of, 10
Girsanov transform, 11
grid computing, 16
HDTV, 15
Huynh, Chi Ban, 29, 30

log-normal distribution, 30
loss
 ratio, 6
Mixed Reality, 17
moment
 matching, 30
 of log-normal variable, 32
multicast, 17
 spanning tree, 19
network
 architecture, 7
 capacity, 6
 matrix, 12
option

basket, 10
rainbow, 10
packet, 6
 loss, 6
path send fee, 13
total, 13
pseudo spot price, 30
push technology, 18
QoS, 5
Quality of Service, 5
rainbow options, 10
Rasmusson, Lars, 5, 7, 11
reliability, 6
remote control of instruments, 16
risk-neutral valuation formula, 10
robotic surgery, 16
send fee, 12, 13
spamming, 18
TCP/IP, 7
Tele-immersion, 17
video-based applications, 15
virtual environments, 17
virtual organization, 16
volatility matrix, 11
References


