Material for lecture #3

1.1 Utility

This document discusses the fundamental issues related to resource allocation in packet networks from the viewpoint of utility. Let us define the utility function as the relationship between the available bit rate (x) and the usefulness of the application from end-user perspective (U). There are, of course, other aspects than bit rate that have an effect on the usefulness, but we mainly ignore them here.

A further complication of this simple model, in which utility is primarily defined as the willingness to pay for the service, is that the customer usually has to reveal her willingness to pay before the actual service occurs. Therefore, if and when the exact characteristics of the service are unknown before the ending of the whole service event, the willingness to pay mainly depends on the previous experiences related to similar service events and on the expectations created by marketing.

Now there certainly are experts arguing that this prediction problem can be solved by well-defined, very accurate Service Level Agreements (SLA)[1]. The assumption is that if the customer can exactly know the service in advance, he or she can make rational decisions whether or not to pay the asked price. By the same token, it seems that the more accurate service definition the better service from customer viewpoint. Nevertheless, the current Internet is mostly working just the other way round: the service specification is loose, while the experienced service level effects mainly to the willingness to pay later (and if not anything else, the customer can change the service provider). As a minimum, we can be sure that the general Internet service model will not quickly change from loose to accurate; actually my belief is, in spite of all the effort to introduce SLAs into the Internet, that in packet networks the prevalent service model will remain quite loose forever because of reasons discussed later in this document.

As a conclusion, in this document utility mainly reflects the user satisfaction that can be assessed only afterward. The monetary value of the possible satisfaction will be realized, if ever, when the customer is making a new decision about the use of the service and about possibly payments. More generally, we may safely assume that if the service fulfills better the user needs, there is a better opportunity for the service provider to collect fares. Whether the service provider can exploit the opportunity is another matter that is related to the overall market situation. In this document we do not address this issue more but just assume that

- there is an unambiguous utility scale in a way that utilities can be added up
- the result is a relevant measure that can be used to compare different service models

Even though these seem to be plausible assumptions, they are not necessarily valid in every case. For instance, it is possible that individuals have certain rights that may

[1] Guaranteed Service within the Integrated Service framework and Expedited forwarding (EF) within the Differentiated Services framework are typical examples of this line of thought.

override the regular utility calculations. Also, it is not clear that the real decision making process of individuals can be modeled by a simple utility model.

1.2 Utility functions

As presented in various papers and elementary books, there are two main function types that depict the usefulness of a telecommunication service (see illustration in Fig. 1):

- A step function when application requires a definite, fixed bit rate, e.g., voice coded by a constant bit rate method.
- Convex functions when application works the better the higher the available bit rate without any significant steps, e.g., file transfer.

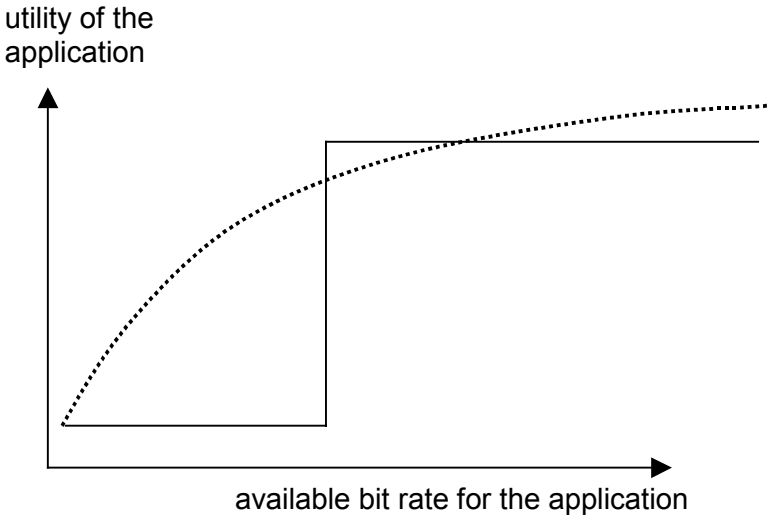

Fig. 1. Convex and step utility functions

With the step function, the height of the step reflects the users' readiness to pay for the service (if the end-user is obliged to pay explicitly for the service) or the usefulness of the application from more general viewpoint (if a larger organization or society pays the service).

The form of utility function has a quite clear connection with the most reasonable way to share the network resources. In a simple homogeneous case with identical flows, either step or convex, the most reasonable way seems to be:

- *With step function*, accept the maximum number of flows that can be served with the required bit rate, and discard all excessive flows.
- *With convex function*, accept all flows and share the resources equally between active flows.

These models reflect the real operation mode of the current telephone and IP networks. Unfortunately, this state of affairs does not lead to any direct conclusion as to the right operation mode in multiservice networks combining essentially differing utility functions.

In order to make our evaluation more concrete, let us assume that the utility function for a flow consist of the following main parts illustrated in Fig. 2.

1. If a user is indifferent to the service, the utility is 0 independent of the available bit rate (x). For instance, when a user is not receiving or calling anyone with your phone, the potential bandwidth available for her is an irrelevant issue, and does not have any effect on the user satisfaction. Obviously, the interest of the service provider is that those users do not consume any network resources.

2. User wants to use a service but the available bandwidth is not sufficient for any meaningful service. In addition the user still remains active in spite of the poor service. For example, if a voice connection applies 10 kbps coding scheme and the available bit rate is 5 kbps; the user will certainly be dissatisfied. The user may still be willing to endure this disagreeable situation for a moment, if she assume that the quality will be improved later. Thus the utility apparently is negative below an application-dependent bit rate, $\boldsymbol{x_1}$ [2].

3. If the available bit rate remains useless longer, the user probably decides to give up and do something else. Although it is not clear in all cases, we may expect that while the utility level also is negative in this case, the utility perceived by the user is higher than in the previous case (note that the user can do something else, perhaps even more useful, whereas in the previous item that is not a likely situation).

4. With adaptive applications there is a region in which the utility is a growing function. We may also assume that in this region the derivative $\partial U/\partial x$ is decreasing but positive, that is, any increase of available bit rate has a positive effect on the utility, but this increase is smaller the higher the available bit rate.

5. Above a certain limit ($\boldsymbol{x_3}$) the utility does not anymore grow. For instance, if the available bit rate exceeds the physical bit rate at the user interface, it is evident that excessive capacity inside the core network has no effect on the utility function.

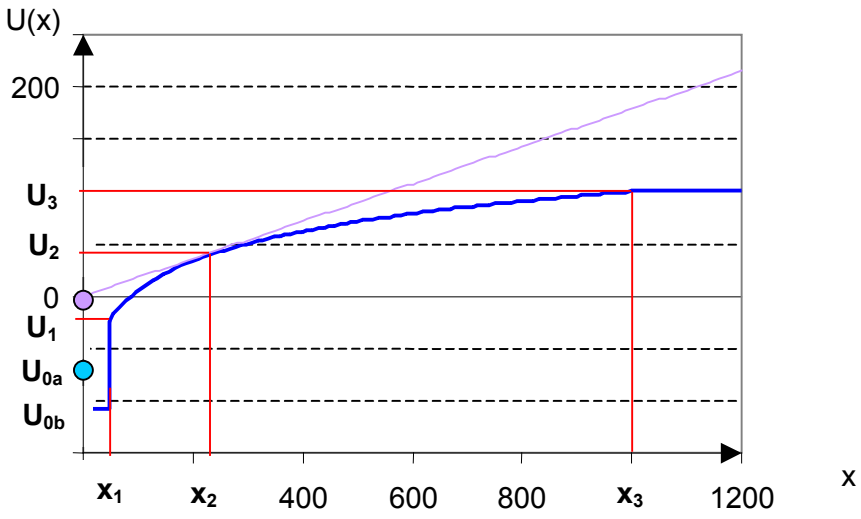

Fig. 2. Utility function for an application
(U_0 = -100, x_1 = 50 kbps, U_1 = -20, x_2 = 224 kbps, U_2 = 40, x_3 = 1000 kbps, U_3 = 100).

Based on these assumptions we may apply the following model for utility as a function of available bit rate:

[2] More accurately, in this case the utility of the application is lower than that perceived by an indifferent user, whether or not the utility is negative or positive is not an essential issue.

A. If the user actively tries to use the service

$$U(x) = U_{0b} \quad if \quad x < x_1$$
$$U(x) = U_3 \quad if \quad x \geq x_3$$
$$otherwise$$
$$U(x) = U_1 + \frac{(U_3 - U_1)[\ln(x) - \ln(x_1)]}{\ln(x_3) - \ln(x_1)}$$

where $x_3 \geq x_1$.

B. If the user basically wants to use the service, but has given up trying to use the service for some reason, utility is

$U(x) = U_{0a}$

The choice of logarithmic function in the middle region is convenient for analyzing purposes, but of course, arbitrary. Whenever possible, the utility function of each application should be considered separately and the most appropriate utility function should be selected.

Note that logarithmic model means that the utility gain is the same if the bit rate is increased from 10 kbps to 20 kbps than if the bit rate is increased from 1000 kbps to 2000 kbps. Although we cannot be sure about the validity of this model, with most applications it certainly is better than a linear model. With a linear model an increase from 10 kbps to 20 kbps is equally useful as an increase from 1000 kbps to 1010 kbps - though possible in some special cases, this kind of situation is very unlikely in reality.

The point (x_2, U_2) in Fig. 2 defines the point in which the utility per available bit rate, U(x)/x, is maximized. Note that if the utility defines directly the user's willingness to pay for the service, bit rate x_2 determines the point where potential revenue per bandwidth is maximized.

1.3 Utility & Traffic Control Principles

Apparently, either an ordinary circuit switched network or a packet network with Connection Admission Control (CAC) is a reasonable approach with a step function, whereas TCP/IP without any admission control works quite well with a convex utility function. Note, however, that because the resulting resource sharing of TCP depends quite strongly on the round trip time of each connection, TCP do not actually divide resources equally (see e.g. "Resource pricing and the evolution of congestion control" by Gibbens, Richard J. and Kelly, Frank P. at http://www.cs.ucl.ac.uk/staff/d.papagiannaki/bibliography.html.) Nevertheless, we may still argue that the *objective* of TCP is to share resources equally among all active connections.

So far, these remarks are relative clear and well known. But how should a network work in more complicated situations? Note that because complexity apparently is one of the most prominent characteristics future networks, it should not be neglected; any model that does not work appropriately in very complex situations is questionable even though it might be optimal one with some specific traffic or service assumptions. Now let us try to proceed from the simplest cases towards more convoluted cases.

The simplest complication is to keep the form of the utility function U=f(x) unchanged but to have different weights for different flows (*i*):

$$U(x)=U_i*f(x)$$

where f(x) is exactly the same for each flow.

The optimum sharing of resources may appear a straightforward task. With a step function the optimum seems to be that resources are given for flows starting from the flow with the highest U_i until the total capacity is filled. But should a new attempt with high utility replace the flow with the lowest utility or should the new request wait until enough resources are released? It seems that there is no unequivocal answer.

In packet networks where resources are, after all, allocated packet by packet basis, it is an easy and attractive approach to serve always packets belonging to the most important flows[3]. However, it is not clear whether this is the optimal strategy because it may be reasonable to give some extra priority for existing flows (it is very annoying if a voice call is ended abruptly).

Then with unequal bit rate requirements (e.g., 10 kbps, 50 kbps and 400 kbps) the design of the system becomes quickly cumbersome. Finally, one flow may have several bit rate thresholds with different utility levels, and the utility function of a variable bit rate flow varies over time.

Similarly, there are a huge number of potential convex utility functions without any universally applicable rule how to divide resources efficiently and fairly. With convex functions the exactly optimal sharing depends also on the form of utility function, f(x), in addition to the weights, U_i. With one specific utility function type, logarithmic, the optimal principle is to divide the capacity proportional to the weights, but that is not a general result with all convex functions. For instance if the utility function is of the following linear form (a limit case of convex function):

$$U(x) = a_i + b_i*x$$

then the optimal strategy is to give *all* resources to the flow with the largest b_i, regardless of the needs of all other flows.

Finally, convex and step functions can be mixed in the same system. Unfortunately, there is not available any clear rule how to share the resources between different type of utility functions - actually, the opinions about this issue vary prominently even among the specialists of this area.

Nevertheless, there are some theoretical studies that try to solve the optimization problem in quite general cases. Though it seems possible in a theoretical arrangement to solve complex cases, in large networks with dynamic traffic conditions the only practical solution is to use some straightforward approximation that does not take into account all the details of every individual utility function. Finally, even though it might be possible, with the huge processing power of current computers, to come close to the optimal solution, in practice there is no exact knowledge about the real utility functions of each user and flow.

1.4 Utility vs. QoS

The previous discussion concentrated on the question how resources should be shared if the utility function is known for each flow. However, if we look at the majority of research papers dealing with the question of resource sharing they are based on QoS evaluation rather than utility evaluation. If and when the objective of

[3] Note that there could be reasons to allocate resources in larger clusters than one packet, for instance, over radio links.

advanced resource sharing is to maximize the total utility, this is somewhat strange situation, because it is not at all clear that any kind of QoS maximization produces high utility. Figure 3 illustrates the situation when packet loss ratio is used as QoS measure.

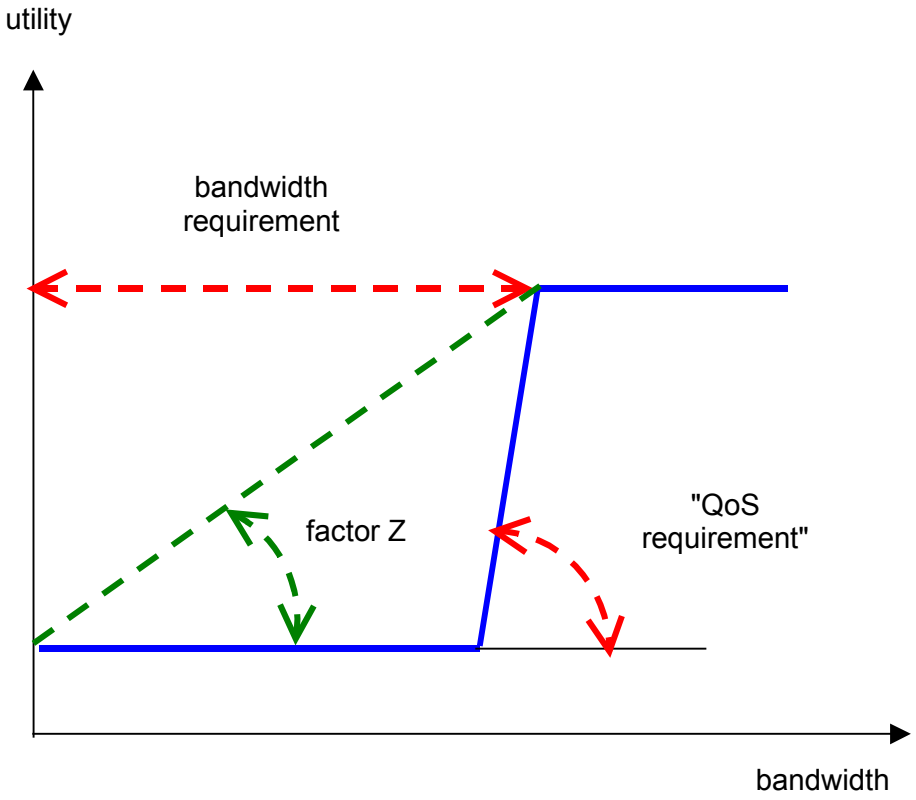

Fig. 3. Relationship between utility and QoS

The traditional way of thinking QoS does not take into account the size of the step, just the steepness of the slope - and that is a big mistake. Instead of QoS the key issue from the viewpoint of reasonable use of resources is the factor X in Fig. 2. But what is factor Z? If you do not know the answer think about it before reading my explanation on the next page.

Factor Z

The horizontal axis is the bandwidth requirement, e.g. 100 kbps, that is able to generate certain utility increase (e.g., 0.001€/s). So the factor Z defines the increase of utility generated by a bit, for instance:

$$Z = (0.001€/s) / (100\ kb/s) = 10\ n€/b$$

Hence the factor Z appears to be a relevant quantity to measure the average importance of a bit (or byte). Now it is of great importance to notice that the slope of the step (traditional QoS requirement) has no direct effect on the importance defined in this way.

The only situation in which the "QoS slope" is relevant is when the available bandwidth is exactly in the region of the slope - but this is very improbable situation if the slope is steep, that is, with strict QoS requirement. With very high probability the available bandwidth is either below or above the slope region, and consequently, the steepness of the curve is irrelevant for resource division.

Let us look at a simple example with two sources, A1 and A2, with parameters shown in table below (see also Fig. 4).

	A1	A2
U_{min}	0	0
U_{max}	1	2
x_{min}	0.98	0.8
x_{max}	1	1

A1 represents an application with (relatively) high QoS requirement but rather low utility, while A2 has the opposite characteristics. To express these figures by conventional QoS framework the allowed packet loss ratio for A1 is something like 10^{-3} (which entails 5% reduction in the utility), and respectively, 10^{-2} for A2. For simplifying purposes the bandwidth requirement is the same (1) for both applications. Now the primary question is how the resources on a bottleneck link should be divided between these applications.

Because both applications possess step-like utility functions, it is quite evident that for a given available bandwidth the optimal strategy is to give priority either for A1 or A2 in a way that either A1 or A2 gets bandwidth 1 if possible and the other one gets the remaining bandwidth. However, it is not evident that the priority is independent of the available bit rate.

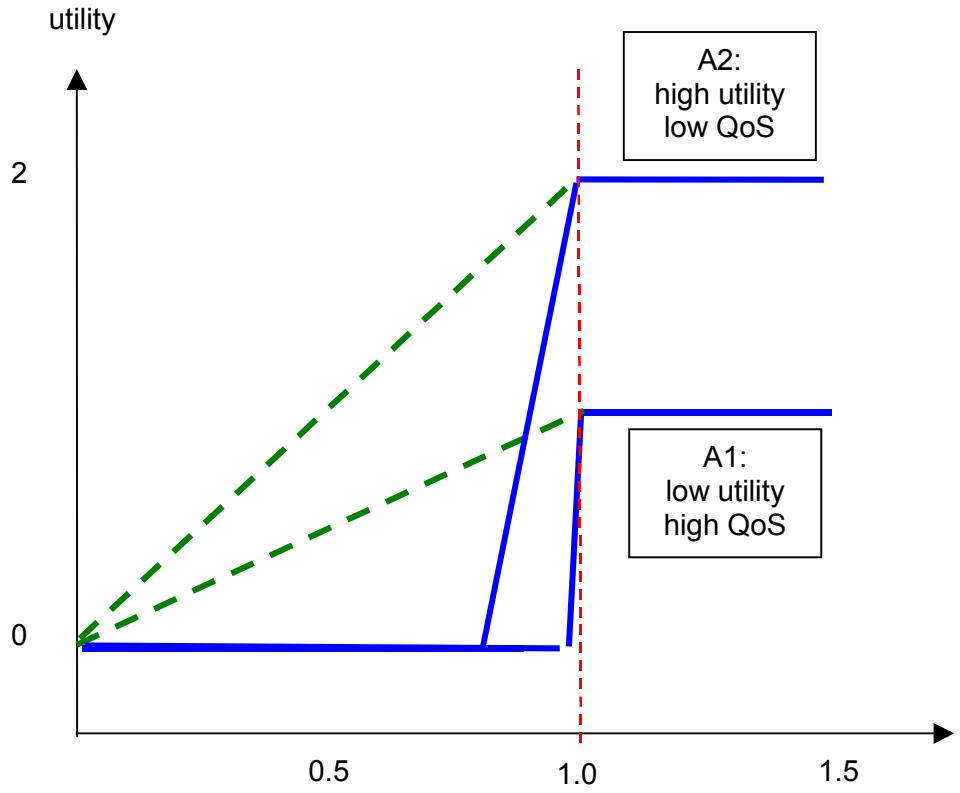

Fig. 4. Two applications with differing QoS requirements and utility values

Because of the simplicity of case, a perfect evaluation is easy. The results are shown in Fig. 5. Three resource division approaches are analyzed:

- A1 gets always higher priority because of higher QoS requirement
- A2 gets always higher priority because of higher utility
- the system selects the optimal strategy

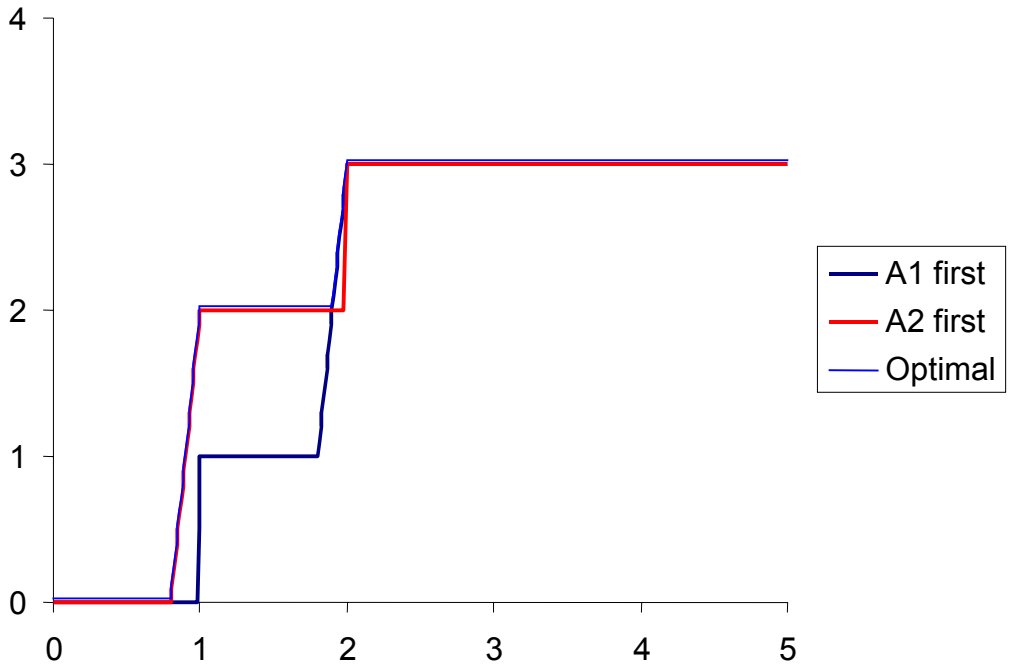

Fig. 5. Total utility with 3 different resource division approaches

What is the optimal, at least in theory, is that in the region from 0.8 to 1.9 it is better to give priority for A2, and in the region from 1.9 to 2 it is better to give priority for A1. Now if we assume, for instance, that the available bandwidth is an evenly distributed random variable between 0 and 5, we get the following average utilities:

always priority for A1: U = 2.04

always priority for A2: U = 2.24

optimal strategy: U = 2.25

An advanced reader may argue that the reason for this result is the unrealistic assumption related to the available bandwidth. Therefore, let us check a somewhat more realistic case in which the available bandwidth is a normally distributed random variable with mean of 2.5 and variance of 1. These assumptions give the following results:

always priority for A1: U = 2.59

always priority for A2: U = 2.74

optimal strategy: U = 2.76

Actually there is not much difference to the first example.

It is left for the reader to consider the question how the result of this example is changed if the packet loss requirement of A1 is changed from 10^{-3} to 10^{-6}.

Questionaire #2 Utility per Application

Environment: Mobile network with advanced terminal with adequate display and camera if needed

Let us fix our utility scale by defining that the utility per minute for GSM-quality voice is 100 (that could be something like 0.10 €/min, but note that only the ratios are here important). Then we can assess the utility curves for other applications by comparing them with this standard value. Further let us assume that the utility of an application can be defined purely by the instantaneous available bit rate, in a way that the short-term utilities can be added up. For assessment purposes let us assume that the available bandwidth for the application is constant always when the application needs the network service, and the packet loss and jitter characteristics are good enough for the particular application.

	minimum useful bit rate	utility/min for min rate	most reasonable bit rate	utility/min for reasonable bit rate	maximum useful bit rate	utility/min for max bit rate
voice						
streaming video						
video phone						
web browsing						
e-mail						
game						

Note: the utility of a GSM-quality voice = 100