ND1115:2001/08

PNO-ISC/INFO/015

ISUP Overload Controls

Network Interoperability Consultative Committee
Oftel
50 Ludgate Hill
London
EC4M 7JJ
UK

http://www.oftel.gov.uk/ind_groups/nicc/

Liability

Whilst every care has been taken in the preparation and publication of this document, NICC, nor any committee acting on behalf of NICC, nor any member of any of those committees, nor the companies they represent, nor any person contributing to the contents of this document (together the “Generators”) accepts liability for any loss, which may arise from reliance on the information contained in this document or any errors or omissions, typographical or otherwise in the contents.

Nothing in this document constitutes advice. Nor does the transmission, downloading or sending of this document create any contractual relationship. In particular no licence is granted under any intellectual property right (including trade and service mark rights) save for the above licence to copy, store and distribute this document and to produce derived copyright works.

The liability and responsibility for implementations based on this document rests with the implementer, and not with any of the Generators. If you implement any of the contents of this document, you agree to indemnify and hold harmless the Generators in any jurisdiction against any claims and legal proceedings alleging that the use of the contents by you or on your behalf infringes any legal right of any of the Generators or any third party.

None of the Generators accepts any liability whatsoever for any direct, indirect or consequential loss or damage arising in any way from any use of or reliance on the contents of this document for any purpose.

If you have any comments concerning the accuracy of the contents of this document, please write to:

The Technical Secretary,
Network Interoperability Consultative Committee,
Oftel,
50 Ludgate Hill,
London,
EC4M 7JJ.

PNO-ISC INFORMATION DOCUMENT 015
ISUP OVERLOAD CONTROLS

.

NETWORK INTEROPERABILITY CONSULTATIVE COMMITTEE

Office of Telecommunications

50 Ludgate Hill

London EC4M 7JJ

0.2 Normative information

All enquiries about distribution reproduction, changes and clarifications should be addressed in the first instance to the Chairman of the NICC/PNO-IG/ISC at the address on the title page.

DISCLAIMER The contents of this specification have been agreed by the NICC. The information contained herein is the property of the NICC and is supplied without liability for errors or omissions.

0.3 Contents

0.4 History

Revision	Date of Issue	Updated By	Description
Issue 1	August 2001	PNO-ISC Editors	Approved by PNO-IG

0.5 Issue control

All	Issue 1	August 2001

0.6 References

[1] ITU-T Recommendation E.412 (03/98), ‘Network management controls’.
[2] ITU-T Recommendation Q.542 (03/93), ‘Digital exchange design objectives – operations and maintenance’.
[3] ITU-T Recommendation Q.543, ‘Digital Exchange Performance Design Objectives’
[4] ITU-T Recommendation Q.764 (12/99) ‘Signalling System No. 7 ISDN User Part signalling procedures’.
[5] PNO-ISC/SPEC006 (BSI/PD6645(2001)) - ‘Interconnect User Part specification’.
[6] PNO-ISC/SPEC007 (BSI/PD6623(2001)) - ‘ISDN User Part specification’.
[7] ‘Memorandum of Understanding: Upgrade of UK interconnect signalling’, Issue 1, Oftel.
[9] ITU-T, Study Group 2, Copenhagen, Temporary Document CPH19: ‘adaptive ISUP overload control’, MJ Whitehead.
[10] US Patent US 5450483: ‘Methods of controlling overloads in a telecommunications network’.
[11] L Kleinrock, ‘Queueing Systems, Volume 2: Computer Applications’, John Wiley & Sons.
[12] European Patent EP 0 729 682: ‘Methods of controlling overloads in a telecommunications network’.
[14] F P Kelly, A K Maulloo and D K H Tan, ‘Rate control for communications networks: shadow prices, proportional fairness and stability’, Journal of the Operational Research Society, 49 (1998), 237-252.
[15] D G Luenberger, ‘Linear and Non-linear Programming’, Addison-Wesley, 1989.
[16] Daisenberger, Oerlerich and Wegmann, ‘Two concepts for overload regulation in SPC switching systems: STATOR and Tail’, Telecommunication Journal, May 1988, pp. 306-313.
[17] Hanselka, Oerlerich and Wegmann, ‘Adaptation of the overload regulation method STATOR to multiprocessor control and simulation results’, Proceedings of the International Teletraffic Congress, June 1988, pp. 43A.4.1-7.
[18] International patent pending WO 99/38341, July 27 1999.
[19] European patent pending EP 0 932 313 A1, July 28 1999.

0.7 Glossary of terms

0.7.1 Abbreviations

ACC	Automatic Congestion Control
ACL	Automatic Congestion Level
DCS	Deferred Circuit Selection
HTR	Hard To Reach
ISDN	Integrated Services Digital Network
ISUP	ISDN User Part
IUP	Interconnect User Part
MTP	(C7) Message Transfer Part
OCL	Overload Congestion Level
TFC	(MTP) TransFer Controlled procedure
TTB	Temporary Trunk Blocking

0.7.2 Definitions

call intent
The initial call attempt in a series of call attempts from a customer to a destination number, all relating to the same call.

customer persistence

The probability that a customer-initiated call attempt repeats after being rejected due to network congestion during call set-up.

source exchange
An exchange that sends calls over a directly connected ISUP traffic route, to another exchange that reports that it is overloaded.

target exchange
An exchange that reports over a directly connected ISUP traffic route that it is overloaded.

1 Introduction

The current signalling system for interconnect between UK operators is IUP [5]. IUP provides an overload control procedure [5] Section 3.2.3.1.6, to protect an exchange from being overloaded by its neighbours (i.e. exchanges directly connected to it by a traffic route).

The UK interconnect signalling system is now moving to UK ISUP [6], in accordance with the relevant Memorandum of Understanding [7]. UK ISUP provides a completely different exchange overload control procedure: Automatic Congestion Control (ACC) inherited from ITU-T ISUP [4]. The latter also specifies Temporary Trunk Blocking (TTB), but it is marked as ‘national use’ in that recommendation and it has not been included in UK ISUP.

The problem with ISUP ACC is that the ITU-T Recommendations are incomplete/imprecise. In particular the specification of what action to take upon receipt of a Release message indicating exchange overload is too vague to guarantee that exchanges that claim compliance to the Recommendations will perform adequately and reliably in overload situations.

PNO-ISC recognised this issue as a potential problem in late 1999, and actioned the PNO-ISC User Part Working Party to setup a Subgroup to investigate ISUP overload controls and provide advice to suppliers and operators on ISUP in the UK.

This document constitutes the findings and recommendations of the Subgroup, and is organised as follows. Section 2 describes likely UK inter-connect ISUP overload scenarios. This motivates Section 3 which defines the end-to-end requirements for UK ISUP overload control. Section 4 describes the IUP overload control. Section 5 describes ITU-T ISUP overload controls (ACC and TTB), and comments on their effectiveness. Section 6 provides what information is known to PNO-ISC on how UK suppliers implement ACC and comments on their effectiveness (without revealing how individual suppliers actually implement ACC). Section 7 proposes a class of ISUP overload control schemes based on controlling call reject rates, and presents results on their behaviour. Section 8 proposes a procedure in regard to ISUP overload controls which UK network operators should follow when negotiating interconnect. Section 9 lists the conclusions of this study. Section 10 lists the recommendations of this study.

Annex A analyses the steady-state achieved by TTB.
Annex B gives information on ISUP message sequences and message lengths used in Section 2.5.
Annex C analyses the behaviour of the class of overload controls described in Section 7 – specifically it characterises their steady-state, and derives necessary conditions for convergence to the (unique) steady-state.
Annex D gives a performance analysis of the Siemens ‘refined ACC scheme’ (Section 6, Solution 4).
Annex E reproduces a slide presentation by Lucent on ISUP ACC.
Annex F reproduces a slide presentation by Siemens on ISUP ACC.

2 UK Network Overload Scenarios

This section identifies the reasons for effective ISUP overload controls, and motivates the UK ISUP overload control requirements defined in Section 3.

2.1 Calling rates/need for load controls

As the following real data (Figure 1) shows, calling rate patterns have largely predictable daily profiles plus frequent, much higher, peaks due to 'events'. Figure 1 shows a 1 in 300 sample of a particular class of calls in a UK operator's network. The displayed sample counts are taken over consecutive 15 minute periods for a month (i.e. 31*24*4 samples for a 31 day month). So a sample count of 500 means an average calling rate over a quarter hour of 500*300/900 = 166 calls/sec. Several consecutive months are drawn as superimposed plots. The important point the figure makes is that the overloads are frequently 4 to 5 times higher than the systematic daily quarter hour peaks. In fact things are worse than that: the vertical axis has been truncated at a sample count of 2500, but has reached (albeit much less frequently) as high as 10,000 (i.e. 3333 calls/sec).

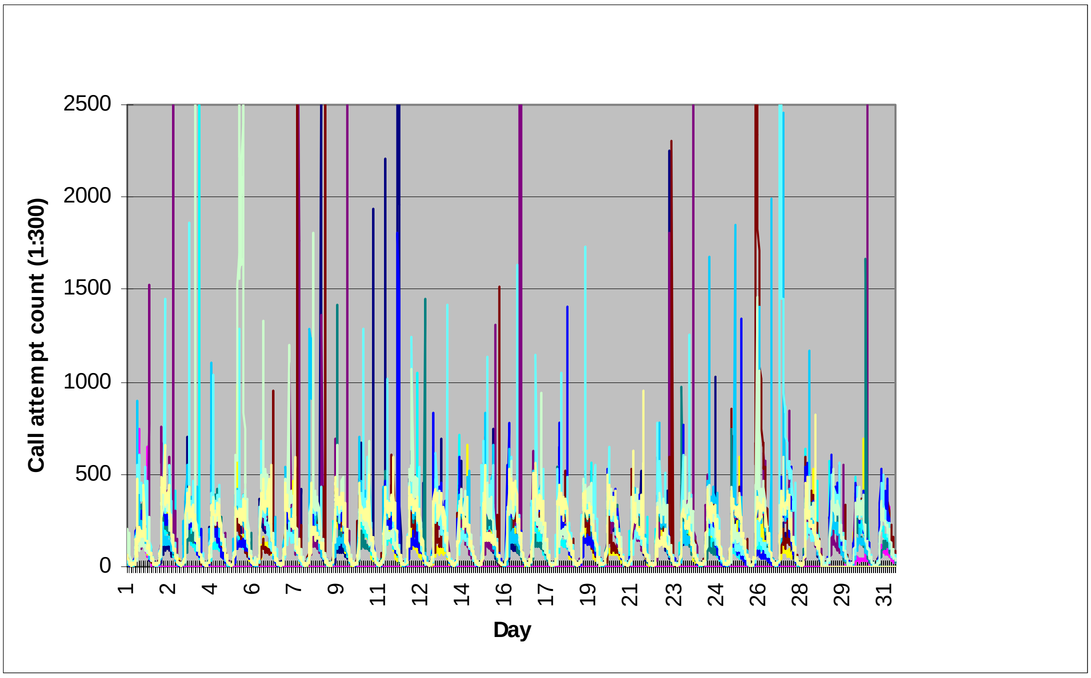


FIGURE 1

Calling rate data

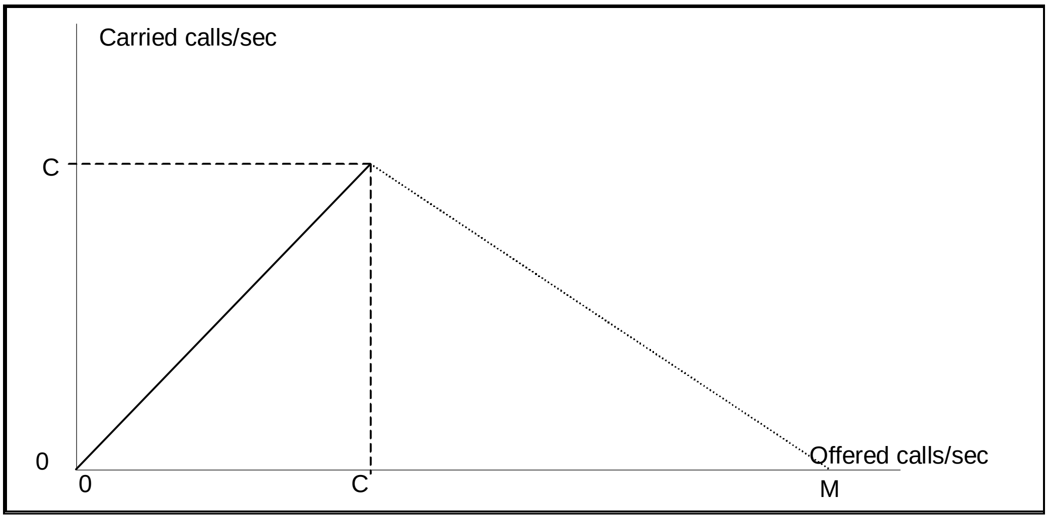
A network operator could provide sufficient capacity to manage calling rate peaks but this is generally uneconomical, and moreover in an overload most calls cannot be terminated successfully because the terminating lines are busy. Therefore controls are needed to manage peaks. Since 'events' are often unannounced, manual controls are not sufficient: fast, automatic controls are required.

2.2 Exchange capacities

In the UK, exchange capacities range from a minimum of roughly 20 calls/sec up to 1000 calls/sec.

2.3 Impact of overloads/need for external load controls

The typical throughput curve for an exchange is shown in Figure 2 – see, for example, ITU-T Recommendation Q.543 'Digital Exchange Performance Design Objectives' [3], Section 3.



Note: The falloff of carried calls/sec beyond 'C' may not be linear.

FIGURE 2

Exchange throughput

As the offered calling rate increases from zero the carried calling rate increases, reaching a maximum of ‘C’ calls/sec. For offered loads less than, or equal, to ‘C’, all offered calls are carried. For offered loads greater than ‘C’ calls/sec the exchange’s internal overload control begins to reject calls, for example to keep processor load at (say) 90%. The processor effort of rejecting calls causes the throughput to decrease. If the offered load is increased sufficiently beyond ‘C’ calls/sec, there then comes a point (at which the offered load is ‘M’ calls/sec) where correct ISUP call handling cannot be guaranteed, and many calls may be rejected. Beyond ‘M’ calls/sec it is likely that internal task queues may overflow, and the exchange may have to restart or roll back to restore itself to a ‘sane’ state. ***This implies the need for external load restriction e.g. at source exchanges causing the overload***.

A good external load restriction scheme will adaptively keep the load offered to an overloaded exchange close to its maximum effective throughput ‘C’. Poor external load restriction schemes over- or under-restrict or oscillate between over- and under-restriction, and cause a large reduction in effective throughput compared to ‘C’, leading to higher chance of call failure.

2.4 Customer repeat attempts

Figure 3 illustrates the effects of high customer persistence on call attempt rates, for an exchange with maximum effective throughput C = 100 calls/sec and whose effective throughput drops to zero at M = 500 calls/sec. Customer persistence can be very high for some events (e.g. 10 call attempts per intent), leading to an ‘explosion’ of repeat attempts when calls are rejected. That causes congestion in other parts of the network. It is therefore essential to maintain high effective throughput under overload.

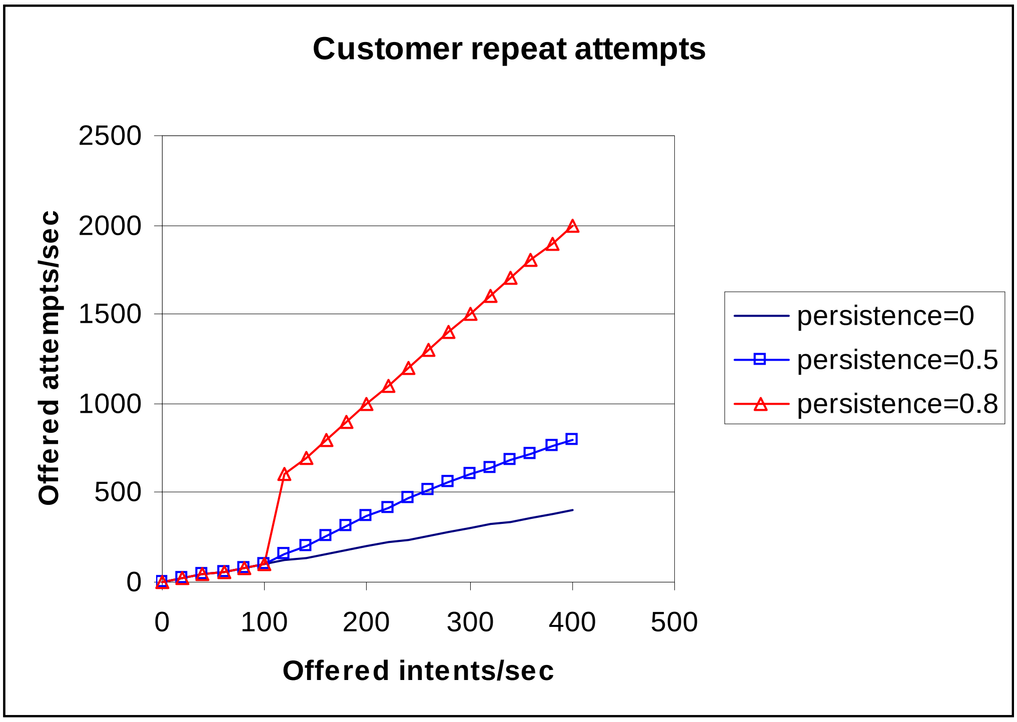


FIGURE 3

Customer repeat attempts

2.5 Traffic route and signalling route capacity

The number of ISUP circuits terminating on an exchange will of course limit the sustained ISUP load that can be offered to it. However, during an overload event, most calls do not terminate successfully, and as a consequence can have very short durations (1-2 sec). It is therefore possible that a target exchange (i.e. the overloaded exchange) with, say, 3600 ISUP circuits, could be subjected to a sustained offered load of 1800-3600 calls/sec.

Assuming each interconnect ISUP traffic route has a 2-link signalling link set, and the total octets per call is 65 (see Annex B), then the link set can deliver $2 \times 0.8 \times \frac{8000}{65} \approx 200$ calls/sec when running at 80% occupancy[1]. It is therefore possible that a target exchange with (say) 6 traffic routes each with a 2-link signalling link set could receive 1200 calls/sec.

It is concluded that, in the absence of an effective ISUP overload control on a traffic route, offered calling rates can be limited by limiting the size of the traffic route (i.e. the number of speech circuits). This is further discussed in Section 10, where short-term guidelines are proposed for discussion.

2.6 Numbers of source exchanges

The number of exchanges generating an overload can vary from event to event. Some events are network-wide (e.g. national TV televotes) with up to 100 source exchanges causing overload; others are more limited in geographical extent (e.g. local radio phone-ins) with a minimum of 1 source exchange causing overload. So any effective overload control must be able to cope automatically with numbers of sources which change from one event to another.

2.7 Calling rate profiles over time

The total volume of calls during an event (or more correctly their calling rate profile over time) will vary from event to event. Most profiles display an initial rapid rise in calling rate (over 1 minute or so) followed by a more gradual decrease back to normal rates (over tens of minutes or an hour or so). In some cases the peak calling rate can be

[1] 80% signalling link occupancy is *not* what a network operator would plan to; rather it corresponds to a high level of overload at which the signalling links might be expected to continue operating adequately.

sustained for an hour or more. An effective control must be able to detect the initial surge of calls very rapidly (within a second or two) in order to protect exchanges from failure due to overload. This means that activation of the control must be rapid and automatic. During an event the calling rate can fluctuate rapidly, and so the control must also be able to rapidly adapt the level of restriction applied.

3 End-to-end requirements for ISUP Overload Control

This section defines the UK ISUP overload control requirements. An ISUP overload control which, when deployed in all exchanges, can be shown to meet these requirements is called *satisfactory* or *adequate* in this document. It should be noted that that these are *end-to-end* requirements on the performance of a satisfactory ISUP overload control, not *per exchange* requirements.

These requirements deliberately do not refer to any individual control (e.g. ISUP ACC, ISUP TTB) – they apply to all controls that aim to protect an exchange from ISUP calling rate overload.

It is important to recognise that the ISUP overload control requirements apply to the ***complete system*** consisting of overload detection, signalling and restriction. Different suppliers are very likely to implement overload detection and restriction in different ways (all of which meet Q.764). It is therefore possible that one implementation could meet the overload control requirements if it is deployed on all exchanges, but that different implementations could fail to meet the requirement if used together. In practice, a network operator may need to consider the latter possibility when assessing how effectively protected his exchanges are in any given network setting.

The UK ISUP end-to-end overload control requirements (in bold italic text, followed by a short explanation in normal text) are as follows:

1. When overloaded, a target exchange shall indicate that it is overloaded with a backward message without undue delay.

This rules out the use of timers at source exchanges to detect that a target exchange is overloaded.

2. When informed of the overload of the target exchange (i.e. offered calls/sec > C), external load restriction at the source exchanges shall automatically cut in and keep the load offered to that target exchange close to C – that is, reach a 'steady state' which maximises the effective throughput of the target exchange.

This requirement stipulates that the control must maximise the target exchange's effective throughput***.***

3. Under overload conditions, the 95%-ile of the response times at the target exchange of calls carried by it, shall not exceed 1 sec in the attained steady state, that is, when the offered load is close to C calls/sec.

The response time is defined as the time from an IAM being admitted by the target exchange to either an IAM being forwarded on (if the target is not the destination for that call), or an ACM being returned (if the target is the destination exchange for that call).

This requirement serves a dual purpose: it ensures that, when the control has settled down to a steady state, customers do not cleardown due to long call setup times (this is necessary to limit customer-initiated repeat attempts); and it limits the round trip delay from source to target to source, which if it became too long might destabilise the control.

4. It shall be possible to configure the controls so that during the initial transient response of the control (i.e. prior to the steady state being reached) the calling rate offered to the target exchange does not exceed (C+M)/2 IAMs/sec measured over any 1 second period.

This requirement ensures that the overload controls react fast enough to prevent the load offered to the target exchange from getting dangerously close to M.(see Figure 2). This requirement implies that when the exchange's internal load control rejects a call, it must *at once* send a backward message indicating that the call has been rejected due to exchange congestion.

5. Requirements 1 to 4 shall be automatically achievable for any scenario characterised by:
- unrestricted customer repeat attempt probabilities
- number of source exchanges in the range 1 to 100

- ***a 'step increase' in the total load offered to the source exchanges (and destined for the target exchange) from 0 to 5C calls/sec or M calls/sec (whichever is the smaller).***
- ***a fast 'ramp increase' in the total load offered to the source exchanges (and destined for the target exchange) followed by a slower ramp decrease***
- ***any distribution of the total offered load among source exchanges***
- ***target exchange capacities in the range C = 20 to 1000 calls/sec.***

This requirement stipulates that the control should be adaptive: that whatever the target exchange's capacity is, and however many sources there are, and however the calling rate is distributed over them, the control will adapt automatically to achieve requirements 1 to 4. The step increase from 0 calls/sec offered load is an absolute worst case for any control to cope with. The ramp-up, ramp-down profile serves to test if a control can adequately track a varying offered load.

6. An overload control should not require new ISUP signalling messages, or new parameters in signalling messages.

This requirement acknowledges that any scheme which necessitated any change to ISUP signalling messages would be *extremely* unlikely to be considered seriously.

4 IUP Overload Control

IUP [5] provides an overload control mechanism which operates as follows. When an overloaded exchange receives an IAM/IFAM marked as a non-priority, non-protected call it returns an Overload message, which causes the sending exchange to defer the availability for selection of the associated circuit for a 2 minute period.

This mechanism is essentially the same load control mechanism as the ITU-T ISUP TTB control. Therefore, the comments made in Section 5.2 on the performance of ITU-T ISUP TTB also apply to this.

5 ITU-T ISUP Overload Controls

5.1 ACC

ISUP ACC is defined principally in ITU-T Recommendation Q.764, Signalling System No. 7 ISDN User Part signalling procedures [4].
Associated recommendations are:
E.412 (03/98), 'Network management controls' [1].
Q.542 (03/93), 'Digital exchange design objectives – operations and maintenance' [2].

The text relating to ISUP ACC in [4] is reproduced in full below, including the section numbering.

> **"2.11 Automatic congestion control**
> Automatic Congestion Control (ACC) is used when an exchange is in an overload condition (see also Recommendation Q.542). Two levels of congestion are distinguished, a less severe congestion threshold (congestion level 1) and a more severe congestion threshold (congestion level 2).
>
> If either of the two congestion thresholds are reached, an automatic congestion level parameter is added to all release messages generated by the exchange. This parameter indicates the level of congestion (congestion level 1 or 2) to the adjacent exchanges. The adjacent exchanges, when receiving a release message containing an automatic congestion level parameter should reduce their traffic to the overload affected exchange.
>
> If the overloaded exchange returns to a normal traffic load it will cease including automatic congestion level parameters in release messages. The adjacent exchanges then, after a predetermined time, automatically return to their normal status.
>
> **2.11.1 Receipt of a release message containing an automatic congestion level parameter**
> When an exchange receives a release message containing an automatic congestion level parameter, the ISDN User Part should pass the appropriate information to the signalling system-independent network management/overload control function within the exchange. This information consists of the received congestion level information and the circuit identification to which the release message applies.

> If the automatic congestion level procedure is not implemented, the automatic congestion level parameter is not acted upon and discarded as normal.
>
> Automatic congestion level actions are applicable only at exchanges adjacent to the congested exchange. Therefore, an exchange that receives a release message containing an automatic congestion level parameter should discard that parameter after notifying the network management/overload control function.
>
> **2.11.2 Actions taken during overload**
> Whenever an exchange is in an overload state (congestion level 1 or 2), the signalling system independent-network management/overload control function will direct the ISDN User Part to include an automatic congestion level parameter in every release message transmitted by the exchange.
>
> The network management/overload control function will indicate which congestion level (1 or 2) to code in the automatic congestion level parameter.
> When the overload condition has ended the network management/overload control function will direct the ISDN User Part to cease including automatic congestion level parameters in the transmitted release messages."

Note should be taken of the crucial sentence in Section 2.11.1 of Q.764: 'If the automatic congestion level procedure is not implemented, the automatic congestion level parameter is not acted upon and discarded as normal.' This could result in an exchange supplier claiming compliance with ISUP ACC even if they do not implement it at all, provided that this sentence is obeyed. This is an unreasonable and unjustifiable interpretation of the text.

It will be observed that Q.764 [4] says nothing about the three key aspects of ISUP ACC:

1. How an exchange should measure/detect its internal overload level
2. How an overloaded exchange should map its load level to the ACL parameter (indicating congestion level 1 or 2) in release messages.
3. How (in sufficient detail) an adjacent exchange should react to receipt of a release message with an ACL of 1 or 2.

ITU-T Recommendation Q.542 [2] Section 5.5.2 (Automatic congestion control system) provides more detail on Items 1 and 3, as follows:

- On measurement/detection Q.542 advises that an exchange should monitor the value of some system quantity (such as the time to perform a complete cycle of operations), and place thresholds on that quantity in order to determine overload level.
- On response to receipt of a release message with ACL 1 or 2, Q.542 recommends that "An exchange receiving an ACC indicator from a congested exchange should activate the assigned ACC controls and start a timer. (The provisional value of the timer is five seconds and is for further study.) Subsequent received ACC indicators restart the timer, when the timer expires, the ACC controls in the exchange are removed."

The fundamental problems with ISUP ACC are:

1. It is not a feedback control. It is therefore not designed to cope adequately with varying numbers of sources of overload, varying offered calling rates, and varying customer repeat attempt behaviour.
2. It is extremely coarse-grained with only two levels of call restriction at source exchanges. When triggered it is likely that ACC will alternate between swamping and starving the overloaded exchange — which is not adequate in a control. There is absolutely no reason why the number of restriction levels should equal (let alone be rigidly tied to) the number of overload states.
3. It is incomplete: most details of call restriction are 'vendor dependent'. This is likely to mean that different vendors will implement overload detection and call restriction in different ways. The details of implementation are all important, however, in determining whether ACC works adequately or not. Generally, it is unlikely to work adequately because of this.

5.2 TTB

ITU-T ISUP TTB is specified in Q.764 [4], Section 2.9.9. It is identical to IUP overload control in all essential respects, namely:

- It uses the Overload message to reject non-priority calls due to exchange overload.
- When an exchange receives an Overload message in response to an IAM/IFAM it defers availability for selection of the associated circuit for 2 minutes (using timer T3).

TTB is marked for 'national use', meaning it may be used within an operator's network, or between operators in the same country.

Two aspects of TTB's performance are assessed here: namely, speed of response to a sudden overload, and steady-state performance.

5.2.1 Speed of response

When all the circuits on a route are busied-out by the TTB mechanism then the calling rate incoming to the overloaded exchange over that route drops to zero, *but not before*. That is, until the *last* circuit on a route is in use or back-busied, that route can continue to offer new calls to the overloaded exchange at a rate only constrained by the capacity of the associated signalling linkset, or the capacity of the source exchange. This leads to an important conclusion: ***TTB will react much more slowly to a surge of calls than ISUP ACC***, because with ACC the overloaded exchange just has to send a *single* Release message with ACL set to 1 or 2 over a route in order to cause the source exchange to reduce its offered calling rate. TTB may therefore fail to protect an exchange from failure due to overload if the exchange has many idle incoming circuits and the offered calling rate is high enough. It is not possible to be more specific than this unless specific scenarios and exchanges are modelled in detail.

5.2.2 Steady state performance

This looks adequate – see the analysis of Annex A. That analysis concludes that, provided the back-busy period is chosen large enough (specifically larger than the ratio of the number of circuits incoming to an exchange divided by the offered rate, M, at which its effective throughput falls to zero), then the total load offered to the exchange reaches a stable equilibrium which is either:

1. less than C if the incoming circuits are the bottleneck (i.e. the total number of incoming circuits divided by the mean duration of calls admitted by the exchange is less than C), or
2. lies between C and M, if the incoming circuits are not the bottleneck (i.e. the total number of incoming circuits divided by the mean duration of calls admitted by the exchange is greater than C).

The steady state performance of TTB is therefore expected to adapt satisfactorily and automatically if the exchange capacity C changes, or if the number of source exchanges changes, or if the route capacity into the exchange changes.

5.2.3 Implementation options

There are (at least) two ways to implement the TTB load restriction algorithm (i.e. deferral of availability for selection of a circuit):

1. Implement the full TTB signalling and restriction mechanisms as defined in Q.764 [4].

   That is, implement the use of the Overload message as well as deferral of availability for selection of a circuit.

2. Use the ACC signalling mechanism together with the Deferred Circuit Selection (DCS) mechanism, which is a scheme that mimics the TTB restriction mechanism. See [6].

   That is, use the Release message with ACL set to 1 or 2[2] to indicate exchange overload, and have the source exchanges defer the availability for selection of the associated circuit for a configured period plus immediately return a Release Complete message to the overloaded exchange to prevent it from timing-out and re-sending the Release.

[2] For the purposes of this proposal, it does not matter whether the ACL is set to 1 or 2.

In performance terms, there should be little discernible difference between these two options, provided that the mechanism which causes a call to be rejected by true TTB is the same as that for true ACC. Whether it is or not can only be answered by each exchange supplier.

A final important point, is that the duration of the back-busy period should be randomised about the configured mean value. This will help to prevent potentially damaging oscillatory behaviour which can occur if the onset of overload is very sudden so that all circuits are busied-out within a few seconds, and therefore get released within a few seconds as well. Taking the back-busy period to be uniformly distributed over the range $\frac{1}{2}h_R$ to $\frac{3}{2}h_R$ (where h_R is the mean back-busy period) should suffice. ***This differs from ITU-T ISUP and IUP which stipulate that the back-busy period shall be 2 minutes exactly.***

Option 2 is preferred and is incorporated in [6].

6 UK ISUP overload controls

6.1 UK ACC implementations

PNO-ISC surveyed all nine UK exchange suppliers represented at PNO-ISC, to request information on how they implement ISUP ACC overload detection and call restriction. This was part of a modelling initiative, to ascertain how different ACC implementations interact in performance terms.

Of the nine suppliers approached (Alcatel, Fujitsu, Lucent, Siemens, Ericsson, NEC, Nortel, Marconi and Nokia) complete replies were received from four. The ACC restriction methods implemented by the four suppliers were of three types, as follows.

> Type 0. No ACC restriction implemented. In this case, as mentioned in Section 5.1 it is unreasonable for an exchange vendor to claim compliance with ITU-T Recommendation Q.764 [4] (and PNO-ISC/SPEC007 [6]).
>
> Type 1. Three internal states: ACL=2, ACL=1, and ‘no congestion’. In any of these three states, the receipt of a Release message with ACL=x immediately puts the restriction algorithm in state ACL=x, and starts (or restarts) a timer. Expiry of the timer immediately causes transition to the ‘no congestion’ state. The level of restriction applied depends only on the state: no restriction in state ‘no congestion’, a configurable level in state ACL=1, and a configurable level in state ACL=2.
>
> Type 2. Three internal states: ACL=2, ACL=1, and ‘no congestion’. In any of these three states, the receipt of a Release message with ACL=x immediately puts the restriction algorithm in state ACL=x, and starts (or restarts) a timer. Expiry of the timer in state ACL=2 causes immediate transition to the state ACL=1 and the timer is reset. Expiry of the timer in state ACL=1 causes immediate transition to the state ‘no congestion’. The level of restriction applied depends only on the state: no restriction in state ‘no congestion’, a configurable level in state ACL=1, and a configurable level in state ACL=2.

The Type 2 scheme is very similar to the Type 1 scheme except that upon expiry of the timer in state ACL=2, transition is to the state ACL=1, not to the state ‘no congestion’.

These sorts of ISUP ACC restriction schemes are arguably what would naturally be implemented guided by the relevant ITU recommendations (Q.764 [4], Q.542 [2], and E.412 [1]). As revealed in Section 6.2, ***they do not meet the UK ISUP overload control requirements***.

At least one UK supplier has implemented a more adaptive form of ISUP ACC. It uses a form of adaptive proportional discard, in which increases and decreases of the discard proportion are governed by two internal timers and the arrival of Release messages with ACL=1 or 2 set. It is reasonable to expect it to be better (because more adaptive) than schemes of Type 1 or 2, but it is not known if it meets the UK ISUP overload control requirements (Section 3).

6.2 Performance of Type 1 and Type 2 schemes

Performance studies of common ACC implementations have been undertaken independently by Siemens and Lucent Technologies. See Annexes E and F.

The Siemens work has proved that the Type 2 (see Section 6.1) implementation of ACC restriction naturally suggested by Q.764 [4] and Q.542 [2] leads to inadequate performance:

1. The target (overloaded) exchange is alternately overloaded and starved of calls

2. The source exchanges (which apply ACC restriction) tend to become synchronised: all increasing their restriction levels together, and reducing them together (termed the Barn Door effect).
3. Under 5.7 x normal load the successful throughput of the simulated network was halved.

Siemens examined various possible solutions:

1. Increasing or decreasing the timer which governs how long the restriction mechanism stays in state 'ACL = 2 ' and 'ACL =1'. (Neither increasing nor decreasing helped).
2. Changing the discard percentage applied in each of those two states. (No single pair of values works across all scenarios).
3. Increasing the number of ACL levels. (Violates the standard).
4. Discarding the use of timers, and instead smoothing the sequence of ACL values received (i.e.. ACL = 2, ACL = 1 and ACL = 0 - release without an ACL value), and mapping it to one of 8 discard levels (0% to 100% in steps of 12.5%). The specific smoothed estimate studied was: $OCL_{NEW} = \alpha \times OCL_{OLD} + (1-\alpha) \times ACL$. The mean value of OCL should 'converge' to the mean value of ACL for suitable values of α. This does not violate the standard.

Solution 4 is the subject of a Siemens patent – see [18] and [19].

A performance analysis of the Siemens 'refined ACC' scheme is given in Annex D.

The Lucent work independently confirmed the Barn Door effect by showing that it also occurs for a similar restriction mechanism (i.e. use of a leaky bucket in place of proportional discard). It also showed that in the absence of any external load restriction, the internal task queues (and processing delays) at an overloaded exchange can grow uncontrollably.

These two studies have convinced the PNO-ISC that the simplest and most natural implementations of ACC restriction (suggested by the relevant standards) fail to work adequately.

Both studies have also showed how vital it is to model the performance of specific ISUP ACC mechanisms (plus overload detection mechanisms). It cannot be safely assumed that an un-modelled implementation will work at all adequately.

7 ACC schemes based on controlling call reject rates

The UK ISUP overload control requirements (Section 3) require that a satisfactory ISUP overload control should be able to achieve a steady state in which the total calling rate offered to an overloaded target exchange is just in excess of that exchange's capacity C. And, moreover, it should be sufficiently adaptive to achieve that steady state irrespective of the target exchange's capacity, and of the number of source exchanges causing the overload, and irrespective of the load destined for the target arriving at the source exchanges.

This strongly suggests basing the control on the *rate at which the target exchange rejects calls*. In particular, the control should be designed to increase (respectively decrease) the rate at which calls are offered to the target according to whether the reject rate is less than (respectively exceeds) a configured reject rate. Such a control should be able to satisfy the overload control requirements, because it can always adjust the calling rate offered to the target so as to match the target's capacity, however many sources of overload there may be, and whatever the calling rate (before restriction) may be, at the source exchanges.

Figure 4 shows the structure of a class of feedback controls based on measuring reject rates at each source exchange.

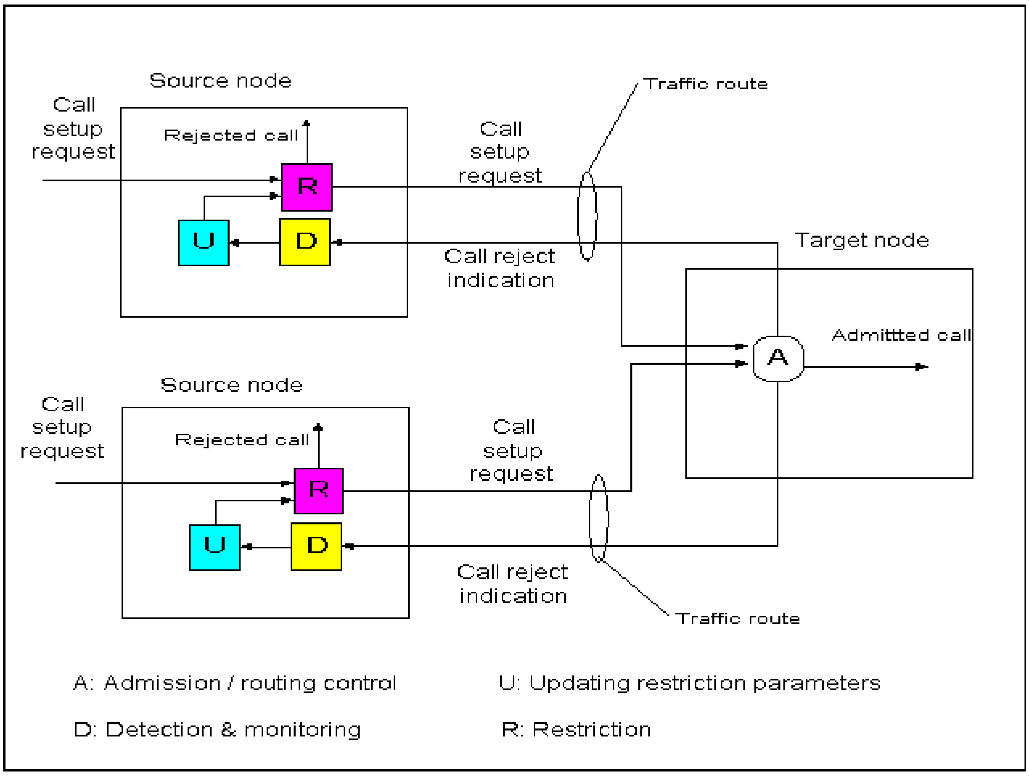


FIGURE 4

Class of overload controls based on reject rates

It is assumed that the target exchange has an admission control (A) which determines whether or not a call should be admitted or rejected by the target, and is linked to the target exchange's internal overload control. The admission process is not part of the proposed control, but is used by it to drive adaptation of the level of call restriction applied at source exchanges.

The proposed overload control has three functional components, labelled D, U and R in Figure 4. There is one instance of D, U and R per traffic route per exchange, and their functionality may be summarised as follows:

- A detection and monitoring process (D) counts calls which are rejected by the target exchange's admission control (A), and measures the reject rate relative to a target rate configured per traffic route at each source exchange.
- A restriction update process (U) updates the level of restriction in response to information received from D — restriction is increased if the reject rate exceeds the detector's target reject rate, and reduced if it is less than it.
- A restriction process (R) thins the incoming demand stream based on its current level of restriction.

This is clearly an incomplete specification, and could not be implemented without defining in detail exactly how the three processes D, U and R actually work. See [9] for a public domain example that is covered by a BT patent [10] & [12]. The above *generic* description of the components D, U and R given above is *not* covered by patent [10] & [12].

Despite this incompleteness, there is sufficient information to analyse two key aspects of the behaviour of this class of controls: namely, steady-state behaviour and conditions that ensure convergence to the steady-state. These are discussed in AnnexesC.1 and C.2 respectively.

7.1 Steady-state behaviour

In Annex C it is shown that the steady state of this class of controls (if it is achieved) is:

1. Unique; and
2. Maximises the effective throughput of the target exchange, whatever the capacity C may be, and however many source exchanges there are, provided that the sum of the individual Detection and Monitoring process target reject rates at the source exchanges is small compared to C.

In addition, Annex C also shows that in the steady state each source exchange gets a share of the target exchange's effective capacity which is *proportional* to the configured target reject rate of its Detection and Monitoring process (D). Thus if the source exchanges have equal rates, then the target exchange's capacity is divided equally between them.

7.2 Convergence to steady-state

In Annex C it is shown that the unique steady state is *always* achieved given some mild constraints on the behaviour of the control instance at each source exchange.

It should be noted that the analysis of Annex C has *not* taken into account the effects of, for example, signalling and nodal processing delays, or the random nature of the offered load. Consequently the constraints must be regarded as necessary for convergence but perhaps not sufficient.

It is of interest to note that these constraints do *not* force source exchanges to implement the controls in exactly the same way. It is not known whether issues to do with the speed of response of the overload controls will necessarily force them closer together.

7.3 Speed of response

7.3.1 Control activation

In order that an overload control of the kind considered in this section should respond rapidly to the initial surge of calls in an overload event, it is necessary that when restriction is activated at the source exchanges the initial level of restriction can be independently configured to be suitably severe. It is not adequate for the control to start restricting at its minimum restriction level.

The use of a separately configured initial restriction level implies the need to be able to configure the control instances at source exchanges so that restriction is not activated by the occasional burst of call rejects. This can be done, for example, by having the reject monitoring and detector process (D) in each source exchange use a leaky bucket to count rejected calls, and only activating restriction if the bucket fill reaches a specific configurable level – this use of a leaky bucket is covered by patent [10] & [12]. No doubt, other mechanisms exist which achieve the same effect.

7.3.2 Speed of adaptation

The restriction level updating process (U) needs to be able to make small changes to the restriction level if the reject rate is close to the configured target reject rate. This is to ensure convergence to the steady-state. Also it needs to make progressively bigger changes to the restriction level as the reject rate departs further from the configured target reject rate. This is to respond rapidly to sudden changes (increases or decreases) in the offered calling rate.

7.3.3 Termination of control

It is natural to have a configurable minimum restriction level (not necessarily zero) at which restriction is deemed to have ended.

If a multiplicative adaptation scheme is used, in which the new restriction level is an adaptation factor times the current restriction level, then the minimum restriction level cannot be zero. This is because in that case if the restriction level reached zero (e.g. due to a mid-event drop in calling rates) then it could not increase thereafter.

When the minimum restriction level is reached at a source exchange, a 'Pending Termination' timer should be started. If the timer expires before any further increase in the restriction level above its minimum value, then control terminates. Otherwise, the timer should be cancelled and adaptation of the restriction level resumes. This timer is essential to prevent the control repeatedly ending and restarting (at its initial severe restriction level) when the ideal restriction level is less than the configured minimum.

8 Procedure for negotiating ISUP interconnect

This section defines the steps that an operator wishing to connect to another operator's network is advised to take. The procedure is important in the absence of any adequate universally implemented ISUP overload control.

8.1 Initial issues to be considered

This section provides an initial list of issues relating to ISUP overload controls that each of the network operators who are involved in negotiating ISUP interconnect are advised to consider.

Questions that two operators negotiating ISUP interconnect shall consider and answer include:

1. Does the other operator's inter-connect exchange implement ISUP ACC overload detection and restriction? If not, what other form of overload detection and control does it offer, if any?
2. Has the behaviour of the other operator's implementation of ISUP ACC been either performance modelled or measured in tests? If not, can you provide any evidence that it will adequately control the demand it sends to my exchange if the latter is overloaded?
3. How does the other operator's exchange respond to an ISUP Release message with the ACL parameter set to 1?
4. How does the other operator's exchange respond to an ISUP Release message with the ACL parameter set to 2?
5. What form of ISUP ACC restriction does the other operator's exchange use? Possibilities include: proportional discard, call gapping, and leaky bucket restriction.
6. Is there a *fixed* restriction level for ACL = 1 (e.g. 50% discard) and another for ACL = 2 (e.g. 90% discard), or can the restriction level adapt?
7. How many internal congestion levels are recognised by the other operator's exchange?
8. What determines when the other operator's exchange ceases applying ISUP ACC restriction?
9. When the other operator's exchange ceases applying ACC does it return to normal operation gradually or suddenly?

Questions that operators might ask their exchange suppliers:

1. Does my exchange have effective internal overload controls? That is, is M several times greater than C? (Refer to Section 2.3 for the definitions of C and M).
2. At what offered calling rate (i.e. M)can correct call handling no longer be guaranteed?
3. At what offered calling rate (i.e. C) is the exchange's effective throughput maximised?
4. At what offered calling rate does the exchange first return Release messages with ACL set to 1?
5. At what offered calling rate does the exchange first return Release messages with ACL set to 2?

8.2 Modelling and testing

Having obtained answers to the questions listed in Section 8.1, each operator should define what overload scenarios (if any) threaten the stability of his network exchanges. If one or other of the operators feels that there is a significant risk of such overloads then he is advised to proceed to the final step: namely performance modelling of the control options at his disposal and testing of them.

Specifically:

1. The scenarios of interest need to be performance modelled in collaboration with the relevant exchange suppliers in order to find a suitable set of overload control parameter values.
2. Additionally test measurements need to be carried out to check that those values do actually work adequately.

The performance modelling step should model the following factors:

1. The origination of ISUP calls at source exchanges destined for the target exchange,
2. The seizure and release of circuits on routes between source and target exchanges,
3. ISUP signalling delay from source exchanges to target exchange,
4. Target exchange's internal overload detection and call rejection mechanism,
5. The way the source exchanges severally implement ISUP overload control; and
6. The message flow between sources and target at call set-up and call clear, in order to capture the total load on the target.

The target exchange supplier will know exactly how his exchange detects internal overload, and maps the level of overload to the ACL parameter in Release messages. The source exchange suppliers know exactly how they implement ISUP overload control upon receipt of a congestion indication from the target.

Because performance models can rarely capture all the relevant details, it is essential to do overload control tests. These should exercise the feedback overload control at realistic levels of offered IAMs/sec to the target exchange, because the performance of such feedback controls depends on the absolute offered rate.

8.3 Interconnect rules in the absence of adequate ISUP overload controls

It may be that having gone through the procedures proposed in sections 8.1 and 8.2, it is agreed by the relevant network operators, that no adequate ISUP overload controls are available, or can be configured so as to give adequate protection. In that case, as discussed in section2.5, it may be possible to limit the calling rate that reaches a target exchange by limiting the size of the ISUP routes incoming to it.

The generic network scenario is that the target exchange has:

1. IUP routes to some operators (over which overload control works adequately).
2. Additionally, routes to internal network exchanges in the target exchange's network (again overload control working satisfactorily).
3. ISUP routes to some operators over which there are implementations of ACC which are known to work adequately.
4. ISUP routes to some operators on which ACC is either not implemented or does not work adequately.

The proposal is to limit the calling rate that the uncontrolled (Bullet 4) routes can offer to the target exchange by limiting the total number of speech circuits 'N' on them. The worst case (i.e. highest calling rate from uncontrolled routes) occurs when all calls on them fail (at the target exchange or beyond) and hence are of 1 sec duration or less. The target exchange can then be offered a calling rate which equals N calls/sec.

To protect the target exchange from this calling rate the proposed 'rule of thumb' would be to limit N to not exceed (M-C)/5[3]. (Section 2.3 defines C as the maximum effective throughput of the target exchange and M>C as the offered calling rate at which correct call handling no longer be guaranteed.) ***It is important to note that this total has to be allocated between all routes from exchanges without controls***.

This rule attempts to take into account that the other routes which do have overload controls on them will probably be offering in total a calling rate ideally slightly greater than C.
It was agreed that C and M should refer to BT inter-connect exchanges since the bulk of inter-connect calls enter BT's network. Consequently, C depends on the exchange type, and lies in the range 100 to 200 calls per second, with M being approximately 10xC.

The mean call duration affects the value of N, since it is given by the following formula:

$$\frac{N}{1-p_S+p_S h_S}=\frac{M-C}{5}$$

where p_S = proportion of calls offered to the overloaded exchange which are successful (ie are effective calls), and h_S sec is the mean duration of such effective calls. This formula assumes that ineffective calls have a mean duration of 1 sec. The scenario is that the controlled routes offered C calls per second in total to the overloaded exchange, and that the total circuits available to all uncontrolled routes (N) is limited so that in the steady-state the rate at which they can send calls to the overloaded exchange (given by the right-hand side of the equation, which is just circuits/mean call duration) equals the agreed calling rate allowed to be offered to the overloaded exchange in excess of its maximum effective capacity C. As an example, take C = 150 calls per second, M = 1500. Then depending on the values of p_S and h_S we get the following table of values for the number of E1's.

	h_S = 1 sec	h_S = 10 sec	h_S = 30 sec	h_S = 120 sec
p_S = 0	9	9	9	9
p_S = 0.1	9	17	35	116
p_S = 0.2	9	25	61	223

TABLE 1

[3] Note that this rate would not be attainable if the total capacity of the uncontrolled source exchanges was less than (M-C)/5.

Number of E1s

It is possible that two exchanges with the same values of C and M may react in different ways to an overload because their internal architectures are different. In that case the proposed rule will need to be revised to take this into account.

The implications of this proposal are that:

1. Each operator needs to know the values of C and M for his exchanges.
2. The lack of effective controls on some routes is managed on a per interconnect exchange basis.
3. The limit on the total number of ISUP circuits from exchanges without controls has to be allocated somehow between those exchanges.

A possible alternative way to protect an exchange from calling-rate overload from exchanges without effective ISUP overload controls, is to limit the signalling link capacity in total available to calls from such exchanges. Two operators negotiating inter-connect should consider the following issues before embarking upon this course of action:

1. Can the signalling link capacity be so restricted?
2. What are the implications for resilience to signalling link failure?
3. Will the MTP-level flow controls perform effectively under signalling link overload? [5]
4. Does the solution adversely impact call streams from exchanges which do have effective ISUP overload controls? This might be the case if controlled and uncontrolled exchanges shared the same signalling network.

9 Conclusions

1. Adequate protection of exchanges requires good *internal* exchange load controls and good *external* overload controls (Section 2.3).
2. UK ISUP overload scenarios can vary widely in terms of target exchange capacity, number of source exchanges, and the calling rate they offer to the target exchange. *External overload controls therefore need to be adaptive* (Section 2.6).
3. During an overload event calling rates can fluctuate rapidly. Consequently both internal and external overload controls must be able to adapt rapidly and automatically (Section 2.7
4. The *fundamental problems* with the ITU-T ISUP ACC procedure are that it is *not a feedback control*, is *coarse-grained* and *incomplete*. Implementations based on it are unlikely to work adequately (Section 5.1).
5. The *steady-state* behaviour of TTB *adapts adequately* to changes in target capacity, number of source exchanges etc. (Section 5.2.2).
6. TTB will *react much more slowly* to a surge of calls than a well-designed implementation of ACC (Section 5.2.1).
7. A class of ACC schemes based on *controlling call reject rates* (Section 7) has been shown to have *acceptable steady-state behaviour*. Necessary conditions ensuring convergence to the steady-state have been established. In principle, this shows that different ACC schemes (based on controlling reject rates) can converge to the unique steady-state.

10 Recommendations

1. The UK-ISUP specification should be include the per exchange UK ISUP overload control requirements.
2. The guidance given in Section 3 should be adopted as the recommended end-to-end UK ISUP overload control requirements.
3. The guidance given in Section 8 should be adopted as the recommended procedure for dealing with ISUP overload control issues when operators negotiate ISUP interconnect.
4. The UK should actively encourage the adoption by ITU-T of a single proven class of ISUP overload controls. It should focus work initially on schemes it is believed will meet the end-to-end requirements given in Section 3.
5. For all UK switch suppliers: as a medium-term measure (until a proven implementation of ACC is adopted by the UK exchange supplier) and in the absence of an effective ACC restriction scheme, PNO-ISC recommends that the supplier adopt the deferred circuit selection scheme defined in section 5.2.3 with a randomised timer.

[5] However, the Barn Door effect could apply to these as it does to the ACC implemented as per [4], see Section 6.2.

Annex A: TTB steady state analysis

A.1 Symmetric case

Consider a target exchange receiving calls from N source exchanges, each of which applies TTB on its route to the target. For the symmetric case, it is assumed that the source exchanges are identical in offered calls per second before TTB restriction, that they have identical routes sizes to the target exchange, that they have identical mean successful call durations, and identical back-busy periods.

Let the maximum carried traffic achievable at the target exchange in the absence of any external load controls be denoted by ‘C’, and the offered call/sec at which a exchange’s throughput drops to zero (again in the absence of externally applied load controls) be denoted by ‘M’ (see Figure A-1).

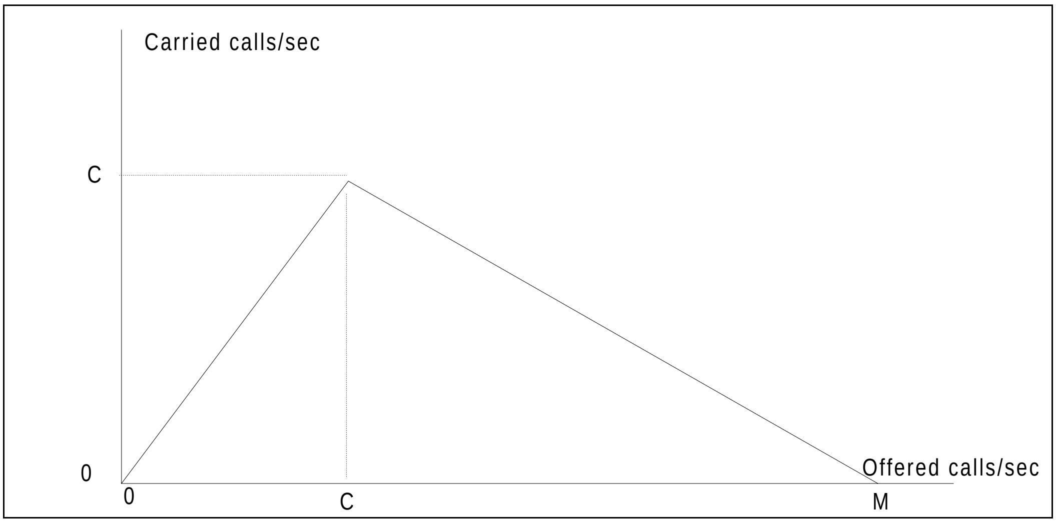


FIGURE A-1

Target exchange throughput

Denote by λ the calls per second each exchange offers to its route to the target. Denote by γ the calls per second admitted onto each exchange’s route and therefore offered to the target. Denote by n the number of circuits on the route from each source to the target. Then the formula for γ is

$$\gamma = \lambda(1 - E_n(A)) \qquad \text{Equation 1}$$

where $E_n(A)$ is Erlang’s loss function:

$$E_n(A) = \frac{\frac{A^n}{n!}}{\sum_{i=0}^{n} \frac{A^i}{i!}} \qquad \text{Equation 2}$$

This is the probability that all n circuits are busy, given an offered traffic of A erlangs - see for example [11, section 1.5].

The offered traffic A is given by

$$A = \lambda((1 - B)h_S + Bh_R) \qquad \text{Equation 3}$$

since the mean holding time of a circuit to the target exchange is $(1 - B)h_S + Bh_R$ where B is the target exchange blocking probability, h_S is the mean duration of a call accepted by the target and h_R is the mean back-busy period. From Figure A-1 it can be deduced that the target exchange loss probability is given by

$$B(\Gamma) = \begin{cases} 0 & \text{if } \Gamma \leq \mathrm{C} \\ 1 - \dfrac{C}{\Gamma}\left(\dfrac{M-\Gamma}{M-C}\right) & \text{if } C < \Gamma < M \\ 1 & \text{if } M \leq \Gamma \end{cases}$$ Equation 4

where $\Gamma = N\gamma$ is the total calls per second offered to the target.

To see this, observe first that for $\Gamma \leq C$ all calls are carried, and hence in that range the loss probability is zero. Next, for $M \leq \Gamma$, no calls are carried, hence in that range the loss probability is 1. Finally for $C < \Gamma < M$, it can be seen that the carried calls/sec denoted by $X(\Gamma)$ satisfies the equation:

$$\frac{C}{M-C} = \frac{X(\Gamma)}{M-\Gamma}$$ Equation 5

due to the linearity of the effective throughput function. But we also know that the carried calls/sec is the offered calls/sec times the probability a call is carried:

$$X(\Gamma) = \Gamma(1 - B(\Gamma))$$ Equation 6

Solving Equation 5 and Equation 6 for $B(\Gamma)$ gives the expression in Equation 4.

These steady-state equations can be solved by fixing values for $C, M, h_S, h_R, N, n,$ and λ and iterating Equation 1.

The equations can exhibit bistability. This is most easily seen by re-casting Equation 1 in the form

$$\frac{\Gamma}{N} = \lambda\left[1 - E_n(\lambda(h_S(1 - B(\Gamma)) + h_R B(\Gamma))\right]$$ Equation 7

and plotting the right-hand and left-hand sides of this equation as functions of Γ. Figure A-2 gives an example (Example 1) for the parameter values

$$\begin{aligned} C &= 280cps \\ M &= 350cps \\ h_S &= 96\sec \\ h_R &= 15\sec \\ N &= 50 \\ n &= 180circuits \\ \lambda &= 8cps \end{aligned}$$

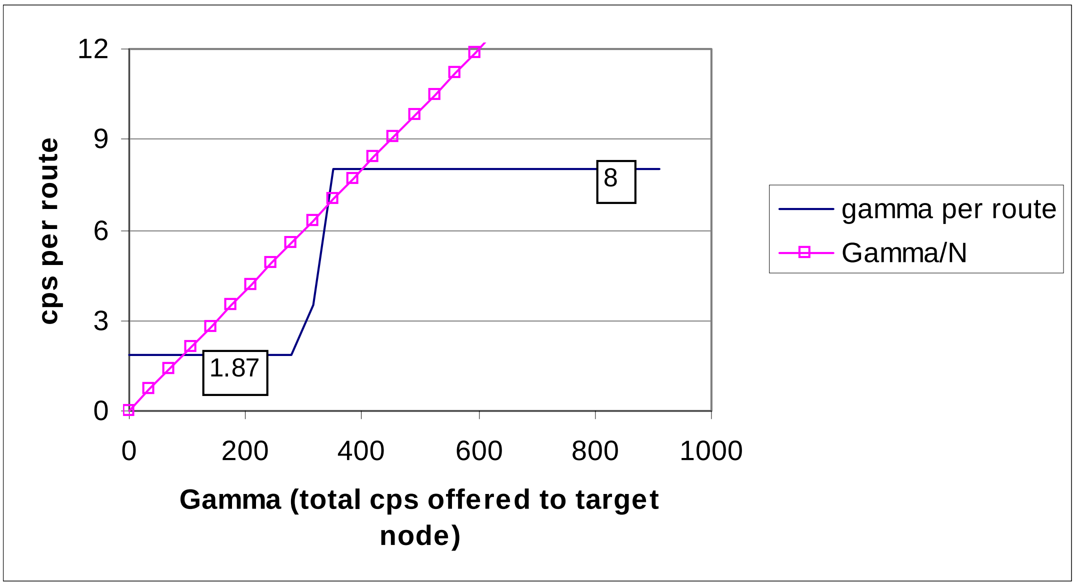


FIGURE A-2

Example 1: Cause of bi-stability

The righthand side of Equation 7 is the unmarked solid curve labelled “gamma per route”. The left-hand side is the curve (marked with squares) labelled “Gamma/N”. It is evident, that for these parameter values, there are 3 points where the two curves intersect. However, by experiment, it is a fact that the middle point is unstable. That is if an initial value for Γ is less than 350 calls per second then the equations converge on $\Gamma = 94$; and if the initial value is greater than 350 calls per second then the equations converge on $\Gamma = 400$. The latter steady state is clearly dangerous since it exceeds the offered rate (350 calls per second) at which the exchange’s effective throughput will have dropped to zero. The other steady state ($\Gamma = 94$) is sensible since it equals the maximum rate at which effective calls could possibly be offered to the exchange given the number of circuits into the target exchange (180*50 = 9000) and the mean duration of an effective call (96 sec).

The reason for this bistability is fairly obvious. If the back-busy duration h_R is less than the duration of admitted calls h_S, then, for high enough total offered rate Γ, a small increase in Γ increases the exchange blocking probability $B(\Gamma)$ and thereby *reduces* the mean duration $h_S(1-B)+h_R B$. This in turn reduces route blocking, and further increases the total calling rate Γ offered to the overloaded exchange: i.e. the system has positive feedback.

Generally, the plot of the righthand side of Equation 7 will look like Figure A-3. To see this we need to consider 3 cases: $\Gamma \le C$, $C < \Gamma < M$ and $M \le \Gamma$.

For $\Gamma \le C$, the exchange loss probability is zero $B(\Gamma) = 0$. So the righthand side of Equation 7 becomes $\lambda[1-E_n(\lambda h_S)]$ and this is approximately equal to (and is bounded above by) $\min(\lambda, \frac{n}{h_S})$.

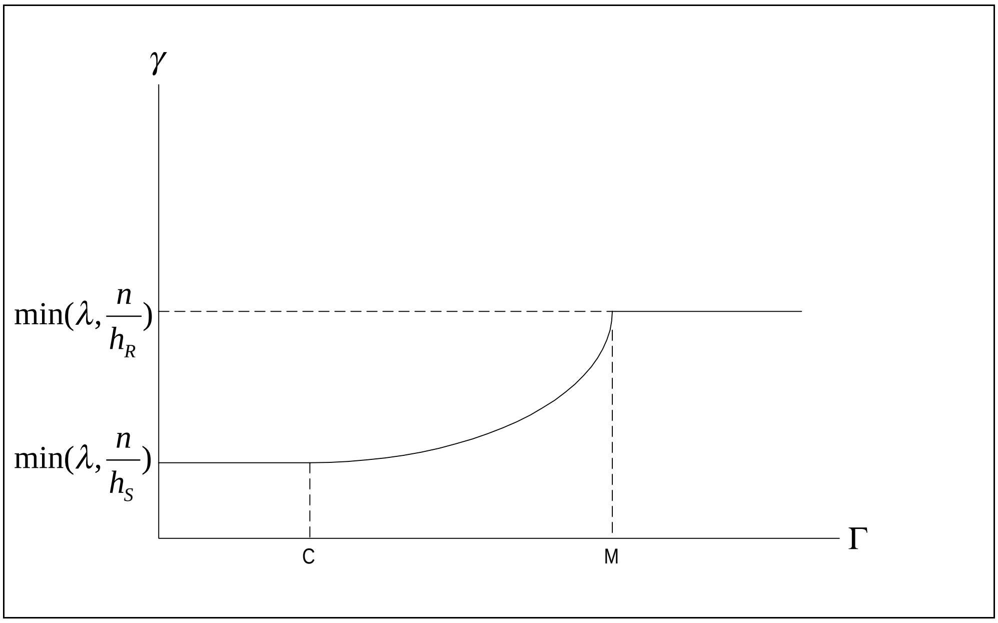


FIGURE A-3

Plot of righthand side of Equation 7

For $M \leq \Gamma$, the exchange loss probability is 1, so that the righthand side of Equation 7 becomes $\lambda[1 - E_n(\lambda h_R)] < \min(\lambda, \frac{n}{h_R})$.

As Γ increases from C to M, the righthand side of Equation 7 will change continuously from $\min(\lambda, \frac{n}{h_S})$ to $\min(\lambda, \frac{n}{h_R})$. Note that if $h_R = h_S$ then the righthand side of Equation 7 is independent of Γ which is a special case of the curve shown in Figure A-3. Note also that if $h_R > h_S$ then the righthand side of Equation 7 will decrease as Γ increases from C to M.

As an illustration of these results, Figure A-2 shows the numerical value of γ for $\Gamma \leq C$ to be 1.87 which is close to the predicted value $\min(\lambda, \frac{n}{h_S}) = \min(8, \frac{180}{96}) = 1.875$. It also shows the numerical value of γ for $M \leq \Gamma$ to be 8, which is in close agreement with the predicted value $\min(\lambda, \frac{n}{h_R}) = \min(8, \frac{180}{15}) = 8$.

It follows that ***in order to ensure there is only a single steady state solution for*** Γ ***and that it is less than M***, we must have

$$\min(\lambda, \frac{n}{h_R}) < \frac{M}{N} \qquad \text{Condition 8}$$

since only then will the righthand side of Equation 7 be less than $\frac{\Gamma}{N}$ at $\Gamma = M$.

This condition will always be satisfied, whatever the value of λ if

$$h_R > \frac{Nn}{M} \qquad \text{Condition 9}$$

This allows h_R ***to be set once and for all provided the maximum value of the righthand of*** Condition 9 ***can be determined***.

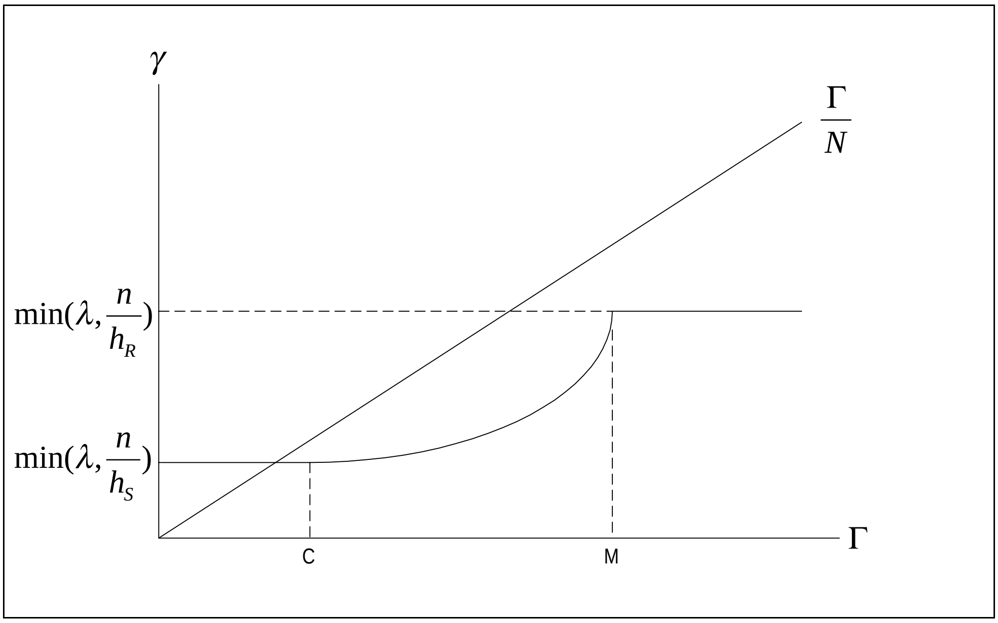

γ
$\frac{\Gamma}{N}$
$\min(\lambda, \frac{n}{h_R})$
$\min(\lambda, \frac{n}{h_S})$
Γ
C
M

FIGURE A-4

Determination of single solution

Given
Condition 9 is satisfied, then there is just a single steady state solution for Γ. From Figure A-4, it can be seen that its value will be $\Gamma = N\min(\lambda, \frac{n}{h_S})$ if $\min(\lambda, \frac{n}{h_S}) \le \frac{C}{N}$, and will lie between C and M if $\min(\lambda, \frac{n}{h_S}) > \frac{C}{N}$.

In either case, the exchange will be offered the maximum calling rate consistent with the values of $\lambda, n \text{ and } h_S$. In the first case, that rate is less than C. In the second case, the exchange will be offered a calling rate which lies in the safe range $C < \Gamma < M$. So, TTB will offer an overloaded exchange as high a calling rate as the trunk groups incoming to it allow, but never in excess of M – the calling rate at which the exchange's effective throughput drops to zero.

Example 2. As an illustration of these results, consider the preceding Example 1. In order to ensure that there is just a single equilibrium point and that it is less than M, we need to take the back-busy period greater than $\frac{Nn}{M} = \frac{50 \times 180}{350} = 25.7$ sec. Setting $h_R = 26$ secs gives the results shown in Figure A-5, which shows that at $\Gamma = M = 350$ the righthand side of Equation 7 is indeed now less than $\frac{M}{N} = \frac{350}{50} = 7$. The numerical value of γ at $\Gamma \ge M$ is 6.74 which is close to the predicted value $\min(\lambda, \frac{n}{h_R}) = \min(8, \frac{180}{26}) = 6.92$

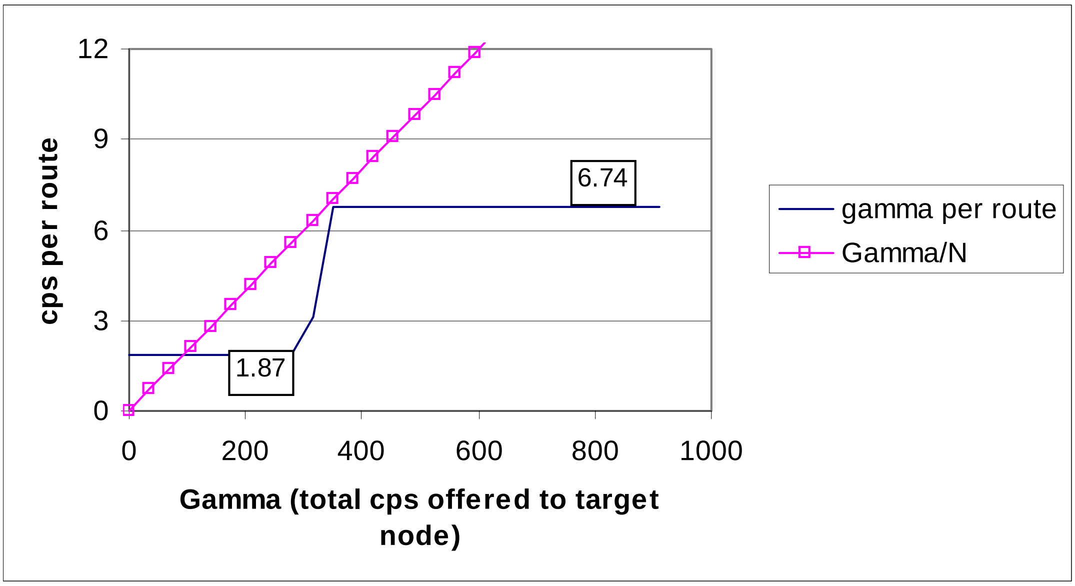


FIGURE A-5

Example 2

Example 3. As a final example, consider Example 1 but with 280*96/50 = 538 circuits per route from each of the 50 source exchanges to the target exchange. This should ensure that a stable equilibrium exists much closer to the target exchange's capacity C = 280 calls per second. First, we remove the high load equilibrium point by taking the back-busy period to be such that

$$h_R > \frac{Nn}{M} = \frac{50 \times 538}{350} = 76.9$$

An 80 sec back-busy period was selected. That should give $\gamma \approx \min(\lambda, \frac{n}{h_R}) = \min(8, \frac{538}{80}) = 6.725$ when the total offered load to the exchange exceeds M = 350. This is very close to the figure calculated by the spreadsheet (6.667) and shown in Figure A-6. Finally, the stable equilibrium should be close to $\min(\lambda, \frac{n}{h_S}) = \min(8, \frac{538}{96}) = 5.6$. This is in good agreement with the numerically calculated value (5.58). This means that the exchange is offered 50*5.6 = 280 calls per second.

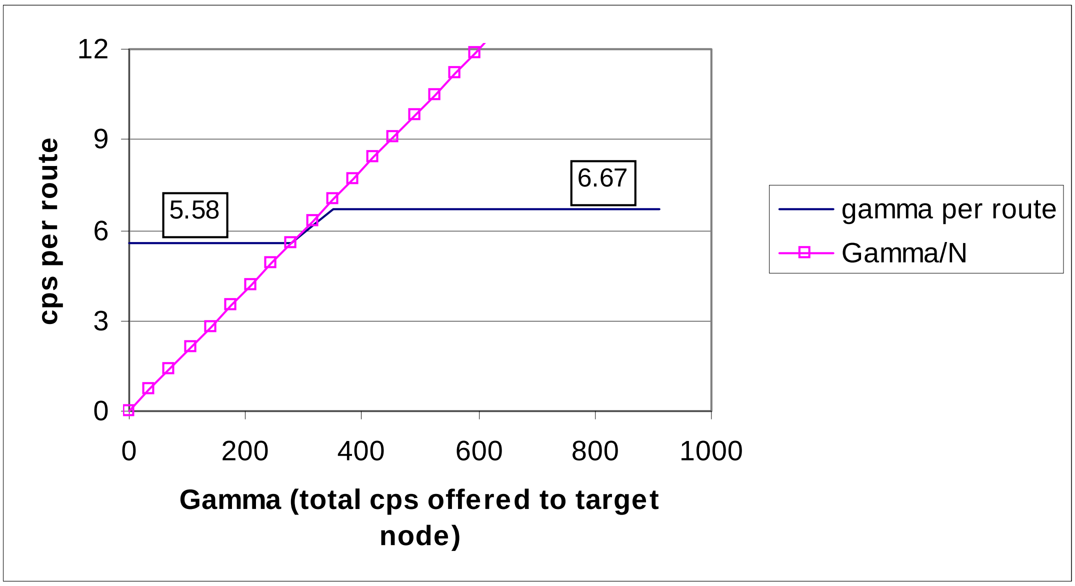


FIGURE A-6

Example 3

A.2 Asymmetric case

The asymmetric case differs from the symmetric one considered in section A.1, in that we allow different source exchanges to have different offered calls per second before TTB restriction (λ_i), and to have different route sizes to the target exchange (n_i). The other assumptions underlying the symmetric case remain unaltered. The steady state equations are now as follows.

The calls per second offered by source exchange i to the target exchange is given by

$$\gamma_i = \lambda_i \left[1 - E_{n_i}(\lambda_i(1-B)h_S + \lambda_i B h_R)\right] \quad \text{for } i = 1 \cdots N$$ Equation 10

and the total calls per second offered to the target exchange is given by

$$\Gamma = \sum_{i=1}^{N} \gamma_i$$ Equation 11

The formula for target exchange loss probability $B(\Gamma)$ remains the same.

A.2.1 Condition for no solution in the region $\Gamma \geq M$

Condition 9 which ensured that (for the symmetric case) there could be no steady-state $\Gamma \geq M$, now generalises to

$$h_R > \frac{\sum_{i=1}^{N} n_i}{M}$$ Condition 12

To see this, we sum Equation 10 over i to give

$$\Gamma = \sum_{i=1}^{N} \lambda_i \left[1 - E_{n_i}(\lambda_i(1-B)h_S + \lambda_i B h_R)\right]$$ Equation 13

If $\Gamma \geq M$, then $B(\Gamma) = 1$, so that the righthand side of Equation 13 is bounded above by

$$\sum_{i=1}^{N} \lambda_i \left[1 - E_{n_i}(\lambda_i h_R)\right] < \sum_{i=1}^{N} \frac{n_i}{h_R}$$ Condition 14

Hence if

$$\sum_{i=1}^{N} \frac{n_i}{h_R} < M$$ Condition 15

which is Condition 12, there can be no steady state solution in the region $\Gamma \geq M$.

A.2.2 Uniqueness of steady state solution

It is possible to give an approximate proof of uniqueness of the steady state vector $(\gamma_1, \cdots, \gamma_N)$ when all the offered rates λ_i are sufficiently large. By ‘sufficiently large’ we mean that

$$\lambda_i \left[1 - E_{n_i}(\lambda_i(1-B)h_S + \lambda_i B h_R)\right] \approx \frac{n_i}{(1-B)h_S + Bh_R}$$ Equation 16

for all $i = 1, \cdots, N$. Equation 11 then becomes

$$\Gamma = \frac{\sum_{i=1}^{N} n_i}{(1-B)h_S + Bh_R}$$ Equation 17

This is essentially the same as Equation 7. The righthand side equals $\frac{\sum_{i=1}^{N} n_i}{h_S}$ if $\Gamma \le C$, and changes continuously to $\frac{\sum_{i=1}^{N} n_i}{h_R}$ when $\Gamma \ge M$. So plotting the left-hand and righthand sides of Equation 17 as functions of Γ looks like Figure A-7, provided that Condition 12 holds so that the righthand side of Equation 17 lies below the left-hand side for $\Gamma \ge M$. It follows that there is just one steady-state value of Γ satisfying Equation 13 and that the individual γ_i (which sum to Γ) are given by the righthand side of Equation 16.

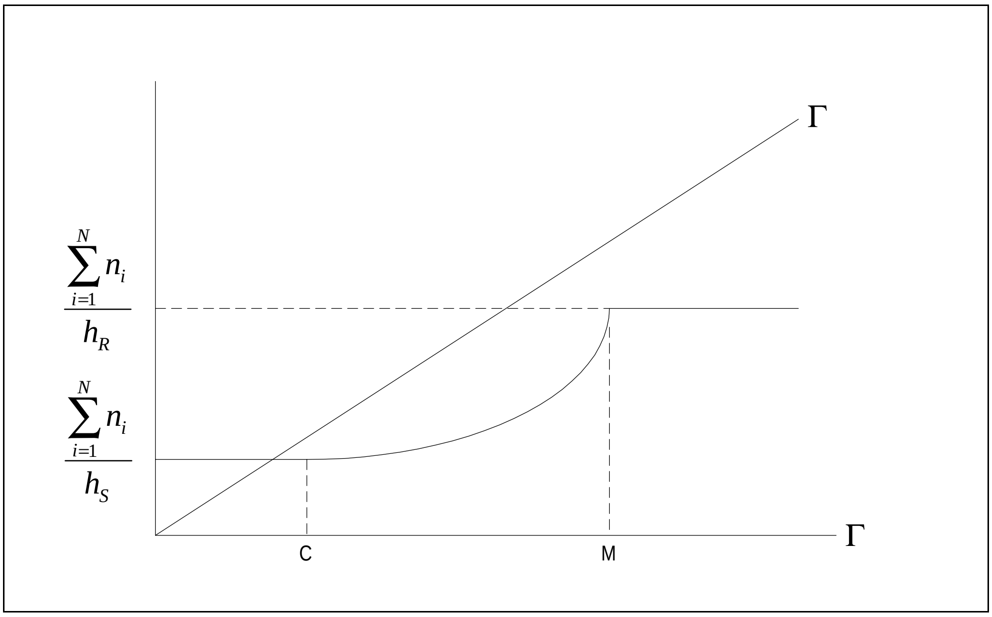


FIGURE A-7

Uniqueness of steady state – large λ_i

Annex B: ISUP signalling messages lengths

A typical signalling message sequence for a successful call incoming from a source exchange to a target exchange is shown in Table B-1, and for a call rejected by the target in Table B-2. The indicated message lengths are given as an example – actual lengths, particularly of the IAM, may differ on specific ISUP interconnect routes.

Message	Octets Source to Target	Octets Target to Source
IAM	50	
ACM		20
ANM		19
REL	15	
RLC		15
Total octets	65	54

TABLE B-1

Successful call

Message	Octets Source to Target	Octets Target to Source
IAM	50	
REL		15
RLC	15	
Total octets	65	15

TABLE B-2

Rejected call

Annex C: Performance analysis of controls based on reject rates

C.1 Steady state analysis

It is simple to analyse the steady-state behaviour of all controls in this class. To do this, some notation is required. Let there be n source exchanges. Let γ_i denote the rate at which the restriction process at source i associated with the route to the target exchange admits calls which are then offered to the target. Let

$$\Gamma = \sum_{i=1}^{n} \gamma_i \qquad \text{Equation 18}$$

denote the total calling rate offered to the target exchange, and let $B(\Gamma)$ denote the probability that the target rejects a call from any source due to internal overload. The rate at which calls from source i are rejected by the target is given by

$$\omega_i = \gamma_i B(\Gamma) \qquad \text{Equation 19}$$

In equilibrium, this reject rate equals the locally configured target reject rate at source exchange i denoted by l_i:

$$\omega_i = l_i \qquad \text{Equation 20}$$

From Equation 18, Equation 19 and Equation 20 it follows that

$$L \equiv \sum_{i=1}^{n} l_i = \Gamma B(\Gamma) \qquad \text{Equation 21}$$

where L denotes the sum of the individual source exchange leak rates.

If the target exchange call rejection probability $B(\Gamma)$ is a continuous and increasing function of the total calling rate Γ offered to the target, there will always be a single unique value of $\Gamma = \Gamma_S$ where L / Γ and $B(\Gamma)$ intersect and which hence satisfies Equation 21, as illustrated in FigureC-1.

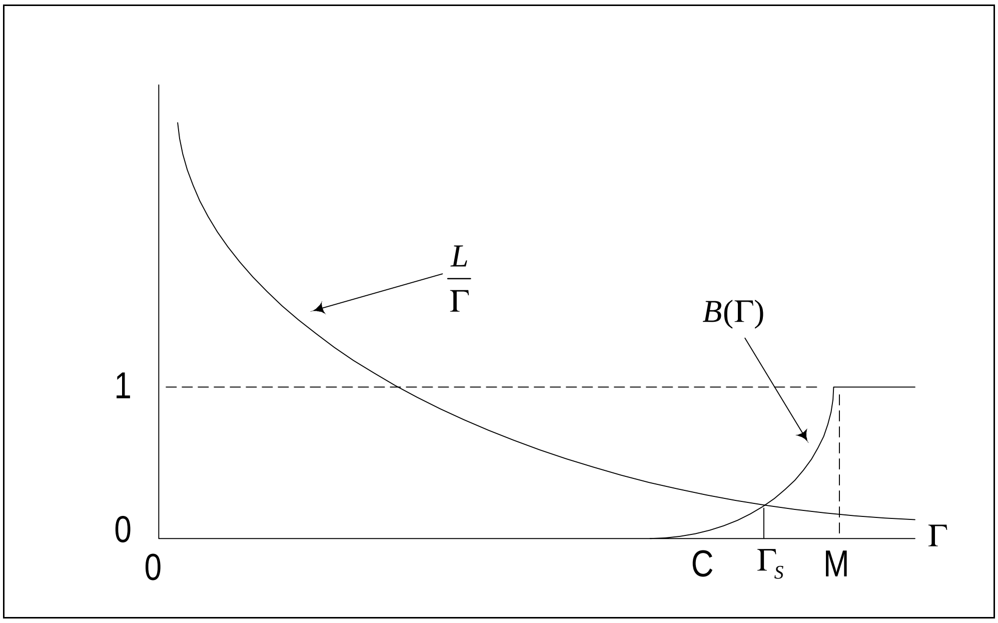


FIGURE C-1

Uniqueness of steady-state solution

It is clear from Figure C-1 that if L is a small fraction of C, then the equilibrium value of Γ will be close to C, since reducing the value of L lowers the curve of L/Γ. ***Thus it may be concluded that this steady-state behaviour of this class of controls (if it is achieved) maximises the effective throughput of a target exchange, whatever the target exchange's capacity C may be, and however many source exchanges there are.***

In addition to this, it can be shown that source i gets a share of the target exchange's effective capacity *which is proportional to its leak rate*. To see this, observe first that the share of the target's capacity that source i gets is given by

$$\upsilon_i = \gamma_i (1 - B(\Gamma_S)) \qquad \text{Equation 22}$$

effective calls/sec.
From Equation 18 and Equation 19 this equals

$$\upsilon_i = l_i \cdot \frac{1 - B(\Gamma_S)}{B(\Gamma_S)} \qquad \text{Equation 23}$$

Summing this equation over i gives the total effective throughput of the target exchange

$$\upsilon = L \cdot \frac{1 - B(\Gamma_S)}{B(\Gamma_S)} \qquad \text{Equation 24}$$

Equation 23 and Equation 24 show that

$$\upsilon_i = \frac{l_i}{L} \cdot \upsilon \qquad \text{Equation 25}$$

That is, source i gets a share of the target exchange's effective capacity which is proportional to its leak rate.

The simplicity of this analysis of the steady-state contrasts dramatically with the complexity of the corresponding analysis of TTB (see Annex A).

C.2 Convergence to steady state

Suppose that the target exchange is subject to high calling rates from each of n source exchanges from some point in time taken to be $t = 0$. We assume that the transient behaviour of the set of overload controls is adequately described by the following set of ordinary differential equations (odes):

$$\frac{d}{dt}\gamma_i(t) = f_i\left(l_i - \gamma_i(t)B(\textstyle\sum_{j=1}^{n}\gamma_j(t))\right) \quad i = 1,\cdots,n \qquad \text{Equation 26}$$

This says that the rate of change of $\gamma_i(t)$ at time t is some function of the difference between the detector target reject rate l_i at source exchange i and the rate of call rejects at source i. The function $f_i(\cdot)$ approximately describes the combined effect of the 3 components (U, D and R) of the overload control instance at source exchange i. It may vary from source exchange to source exchange. The only conditions placed upon it at this point in the analysis are that it is continuous and takes the value 0 only when the reject rate at source i equals the target reject rate at that source. This ensures that in equilibrium (i.e. when all derivatives $d\gamma_i(t)/dt = 0$) we must have

$$l_i = \gamma_i(t)B(\textstyle\sum_{j=1}^{n}\gamma_j(t)) \qquad \text{Equation 27}$$

for all $i = 1,\cdots,n$.

The set of odes given by Equation 26 is almost a special case of the odes considered in [14] as part of that paper's stability analysis of sets of internet overload controls. The only difference between the Equation 26 and those considered in [14] is that that paper considers the case where all the functions $f_i(x) = \kappa x$ for a positive constant κ. It turns out that the stability analysis in [14] still carries through successfully provided only that a mild additional constraint is placed upon the functions $f_i(x)$. That constraint is that $f_i(x)$ is positive when x is positive and negative when x is negative. This is intuitively reasonable, since it just says that the control instance at source exchange i increases its admitted rate $\gamma_i(t)$ at time t if $l_i > \omega_i(t) = \gamma_i(t)B(\sum_{j=1}^{n}\gamma_j(t))$ and reduces it if $l_i < \omega_i(t)$.

Translated to the set of feedback controls given by Equation 26, the method used in [14] to establish stability basically first proves that the function

$$V(\gamma_1,\cdots,\gamma_n) = \textstyle\sum_{i=1}^{n} l_i \log\gamma_i - \displaystyle\int_0^{\sum_1^n \gamma_i} B(y)dy \qquad \text{Equation 28}$$

is strictly concave [15, Section 6.4] on the set where all $\gamma_i > 0$ and hence has just a single local (and hence global) maximum, attained at the point where its gradient vanishes:

$$\frac{\partial V}{\partial \gamma_i} = \frac{l_i}{\gamma_i} - B(\textstyle\sum_{j=1}^{n}\gamma_j) = 0 \quad i = 1,\cdots,n \qquad \text{Equation 29}$$

Then, it is shown that the derivative of V along a solution trajectory:

$$\begin{aligned}\frac{dV}{dt} &= \sum_{i=1}^{n} \frac{\partial V}{\partial \gamma_i} \frac{d\gamma_i(t)}{dt} \\ &= \sum_{i=1}^{n} \left(\frac{l_i}{\gamma_i(t)} - B(\textstyle\sum_{j=1}^{n} \gamma_j(t)) \right) \cdot f_i \left(l_i - \gamma_i(t) B(\textstyle\sum_{j=1}^{n} \gamma_j(t)) \right)\end{aligned}$$

Equation 30

is positive (except where V attains its maximum where ${}^{dV}\!/_{dt}$ is zero) provided that $f_i(x)$ is positive when x is positive and negative when x is negative. So, along any solution trajectory the feedback controls jointly maximise V, and all trajectories must converge to the unique point which maximises $V(\gamma_i, \cdots, \gamma_n)$ which is characterised by Equation 27.

We may therefore conclude that provided each function $f_i(x)$ is positive when $x > 0$, zero at $x = 0$, and negative when $x < 0$ then the overload controls converge globally to the unique steady-state discussed in section C.1. ***This is very important, because it says that different source exchanges may implement the controls in different ways (as characterised by their different functions $f_i(\cdot)$), but convergence to the unique steady-state is nevertheless guaranteed.***

Now, of course, real overload controls cannot be exactly described in this way, because the analysis has not taken into account the effects of, for example, signalling and processing delays, the stochastic nature of demand, etc. Consequently the constraints on the functions $f_i(\cdot)$ must be regarded as *necessary for convergence* of the set of overload controls rather than sufficient. In practice, more detailed modelling would be required.

Annex D: Performance analysis of the Siemens solution (Section 6.2, Solution 4)

D.1 Solution Algorithm

The solution is an enhancement to the standard algorithm: ACLs of 1 and 2 are transferred between exchanges as before. However, traffic is no longer regulated according to the ACL but a refined overload and congestion level (OCL) is introduced that is computed from several previous ACLs. In addition, release messages without ACL are interpreted as if a message with ACL=0 was sent. This makes sense, since no ACL means that there is no overload. Standard ACC does not exploit this valuable information.

The success of such a strategy largely depends on the way the information contained in the ACLs is processed. The idea is to reconstruct a load profile from previous ACLs. This load profile is reflected in the OCL. The OCL must have sufficiently many values to allow for smooth traffic regulations. To stay close to the standard reduction levels of 0, 12.5%, ..., 100% are used.

The OCL is computed not only from the last ACL received at a switch but from several - e.g. the last 20 - ACLs. A weighted sum

$$OCL_{new} = \sum_{k=1}^{n} \omega_k ACL_k$$

computed, where ω_0 is the weight for the most recent ACL and ω_n is the weight for the "oldest" ACL.

Good results were achieved with a weight

$$\omega_k = \frac{\frac{1}{\sqrt{k}}}{\sum_{k=1}^{n} \frac{1}{\sqrt{k}}}, n = 20$$

The resulting OCL has a value between 0 and 2. The mapping to the reduction levels is shown in Table D-1.

Range of OCL	Reduction level
[0, 0.25[	0.0 %
[0.25, 0.5[	12.5 %
[0.5, 0.75[	25.0 %
[0.75, 1.0[	37.5 %
[1.0, 1.25[	50.0 %
[1.25, 1.5[	62.5 %
[1.5, 1.75[	75.0 %
[1.75, 2.0[	87.5 %
OCL = 2.0	100.0 %

TABLE D-1

Mapping of OCL to reduction level

Remember that empty release messages are interpreted as if a release message with ACL = 0 was received.

Hence, the OCL can be smaller than 1 and, in fact, very often is. With the above definition, ω_k decays with growing *k*. Hence, the more recent an ACL the stronger its impact on the computation of the OCL. The effect of different weights can be investigated through simulations. Simulations were also done with other weights, e.g.

$$u_k = \frac{1}{k} \Big/ \sum_{k=1}^{n} \frac{1}{k} \;;$$

u_k decays faster with k than ω_k. That means, that with u_k more influence is accorded to the more recently received ACLs. Choosing an optimal formula for the weights is equivalent with finding the right balance between fast and slow decay of the weights, or the influence between "older" and "newer" ACLs.

Instead of using a weighted sum to compute the OCL one can use the well known recursive formula

$$OCL_0 = ACL_0;$$
$$OCL_n = \alpha\, OCL_{n-1} + (1-\alpha)\, ACL_n$$

Note, that in the recursive formula, the terms ACL_n and OCL_n denote the most recent ACL and OCL, and ACL_0 is the first ACL that has been received.

Computing the OCL with this recursive formula is equivalent to a weighted sum with *all* preceding ACLs. However, the influence of the "very old" ACLs is extremely small. It decays exponentially. Both approaches, the weighted sum and the recursive formula, can be tuned to yield similar results. The recursive formula has the advantage of being less computationally intensive and of requiring less storage space. In the simulations a value of α= 0.9 was found to be advantageous.

Regulation through the OCL leads to smoother traffic variations. However, the information problems cannot be overcome completely, since information is transferred "piggyback" between exchanges and all neighbouring switches of an exchange in overload continue to react almost simultaneously. It would therefore be unwise to disable internal call rejection methods that can be used efficiently for further fine tuning. In the experiments below an internal call control is used like it is described in [16], [17].

Note that the results for standard ACC are without internal call control. However, when internal call control is switched on for standard ACC the results improve only slightly.

With enhanced ACC a network throughput is achieved that is close to that of a network where all switches are working at full capacity but without congestion.

The standardised ACC interface between switches remains unaffected. At the switch itself the OCL mechanism is implemented on top of the standard procedures for the ACL making use of the freedom of choice the standard leaves to the carrier on how to regulate traffic.

A Siemens patent [18],[19] is pending on the refined ACC algorithm described above .

D.2 Simulation results

Simulated was a scenario with 5 nodes (The choice is was made to reduce computing time; experiments with more nodes revealed little change in the results. There is no reason why the results should differ much even with many more nodes). The simulation time was 384 seconds (real time) behaviour; the steady state was reached completely in this time interval. Longer run times were also checked; there were no differences in the results.

In the following the simulation results for the enhanced ACC (using the Siemens solution) are compared to simulation results with standard ACC:

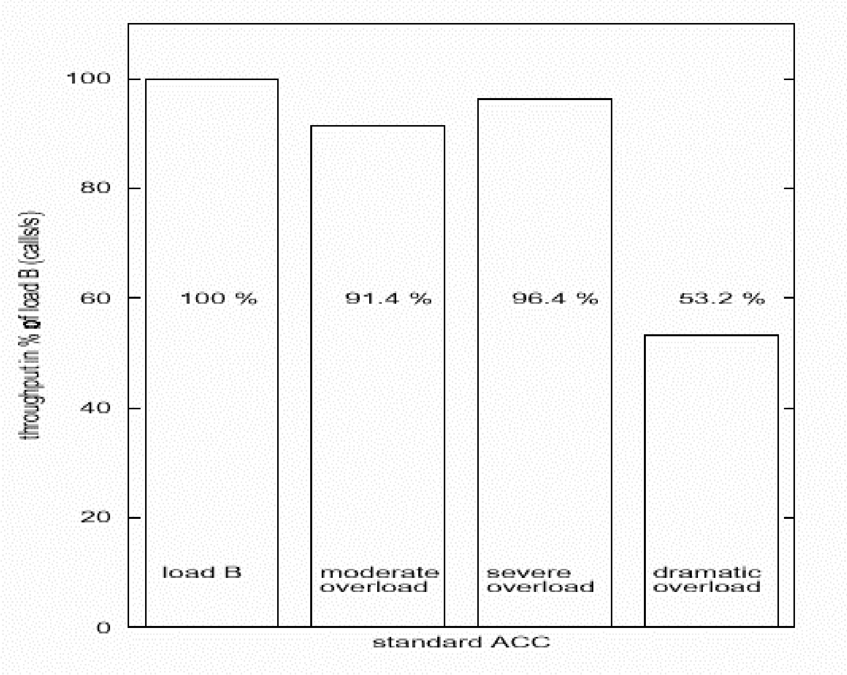


FIGURE D-5

Network throughput for standard ACC

All three overload situations are compared with the network throughput at high load (load B) which corresponds to 100%. With moderate overload the "barn door effect" sets in. Too much traffic is throttled at the neighbouring switches and the network throughput significantly lower than for load B. It increases somewhat in severe overload situations. With dramatic overload each node in the network is in overload so that the node at the centre is not only protected but also throttles traffic to its neighbours. Network throughput slumps.

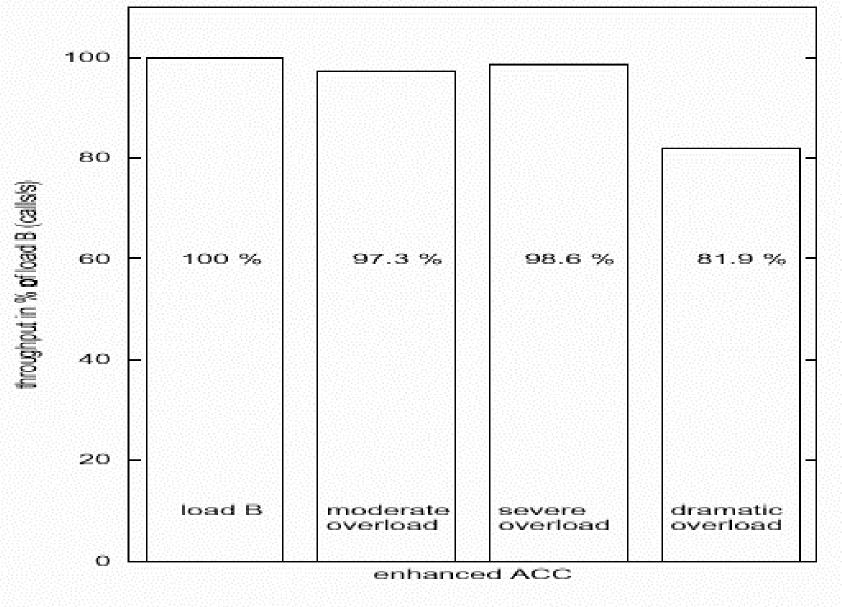


FIGURE D-6

Network throughput for enhanced ACC

Network throughput remains at a very high level with moderate and severe overload. Even in extreme overload situations, when the central node starts to throttle traffic to its neighbours, it remains at a value of over 80%.

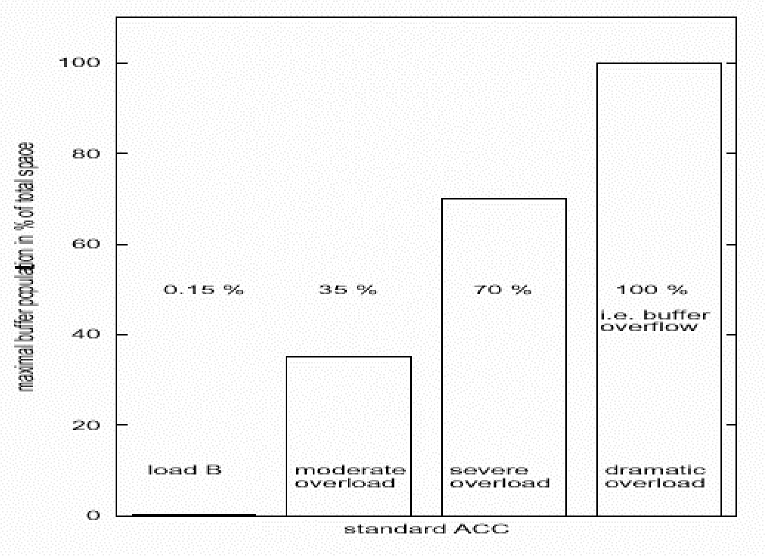


FIGURE D-7

Maximal buffer population for standard ACC

Hardly any entries accumulate in the queue with high load (load B). With moderate overload the maximal buffer population is still acceptable, leaving sufficient spare capacity to deal with additional load fluctuations. It doubles with severe overload reaching a level where one might expect buffer overflows in a long term simulation of several hours. With dramatic overload buffer overflow occurs repeatedly.

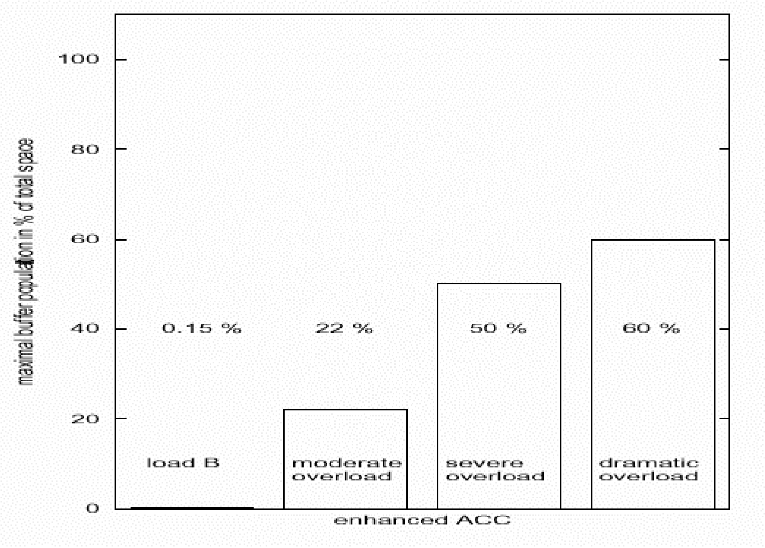


FIGURE D-8

Maximal buffer population for enhanced ACC

Despite the high throughput, the queue length is much shorter than for the standard ACC algorithm. Even for dramatic overload it stays out of the "danger zone" where one might expect buffer overflow in a long term simulation.

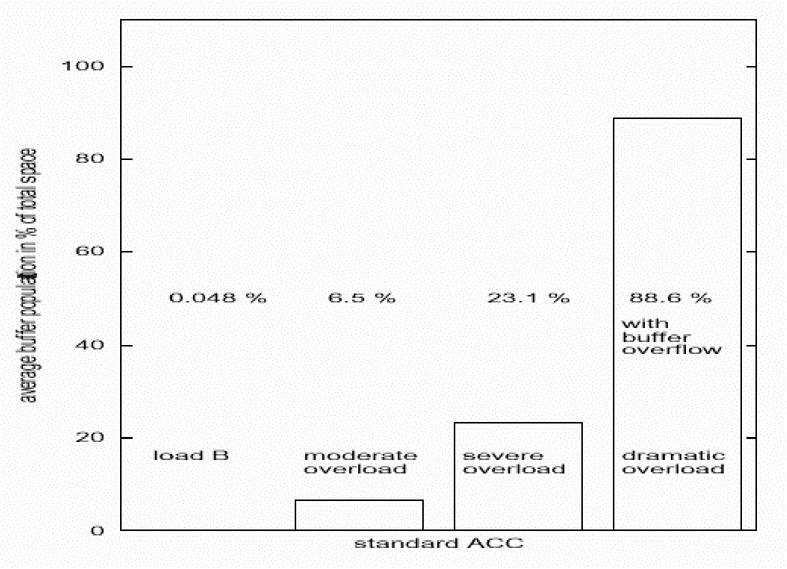


FIGURE D-9

Average buffer population for standard ACC

Comparing the average to the maximal buffer population reveals that the buffer population fluctuates dangerously even with moderate and severe overload.

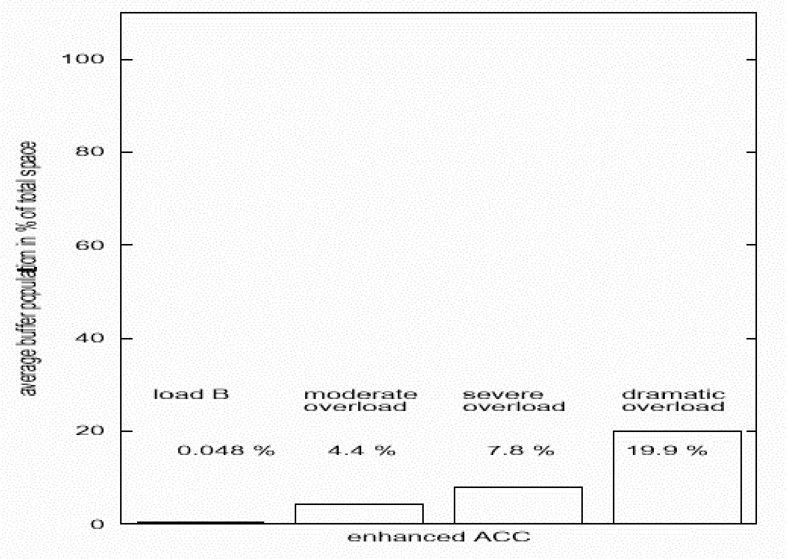
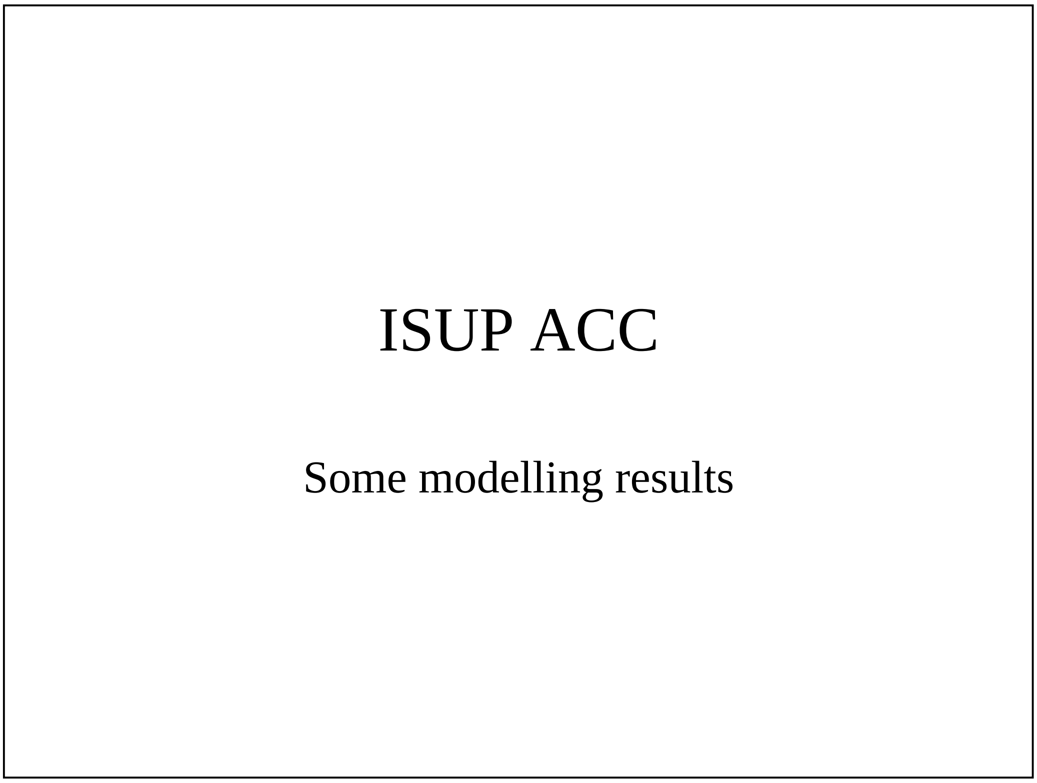


FIGURE D-10

Average buffer population for enhanced ACC

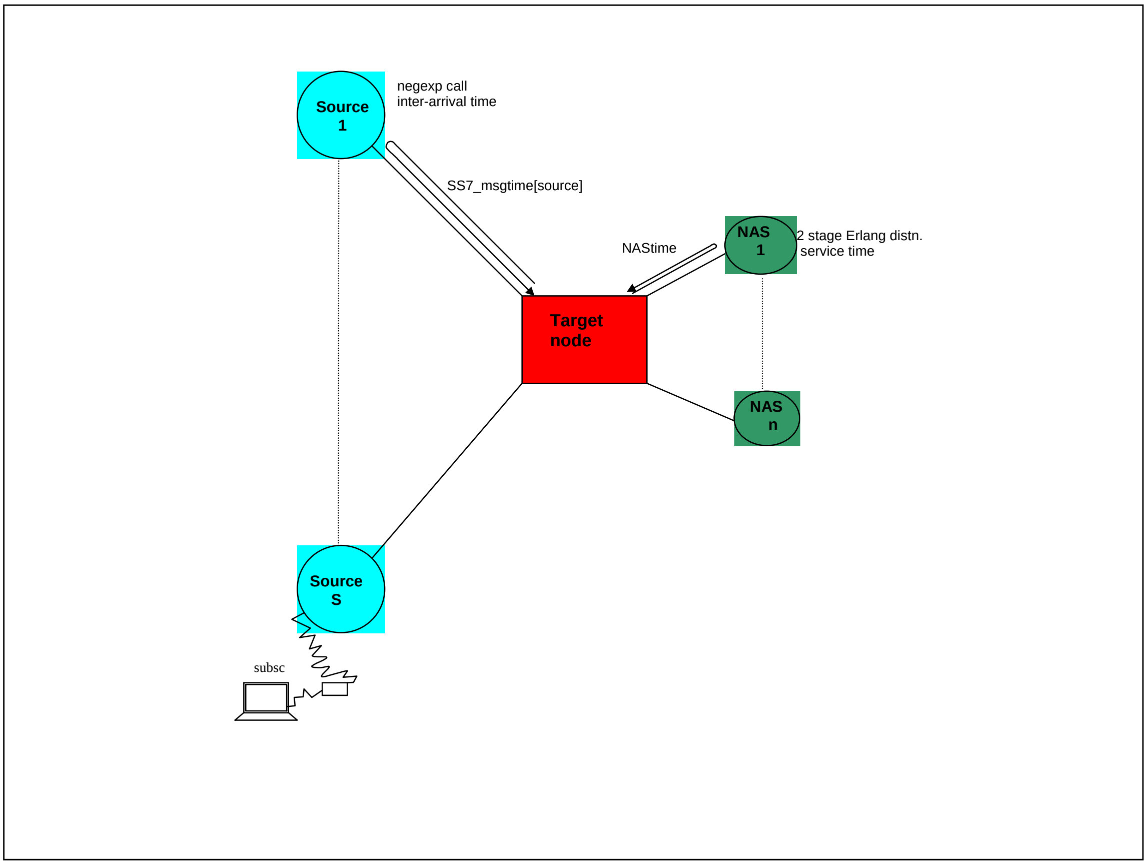
The average buffer population remains low even in extreme overload situations so that one need not fear buffer overflow.

Annex E - Lucent Slide Presentation on ISUP ACC



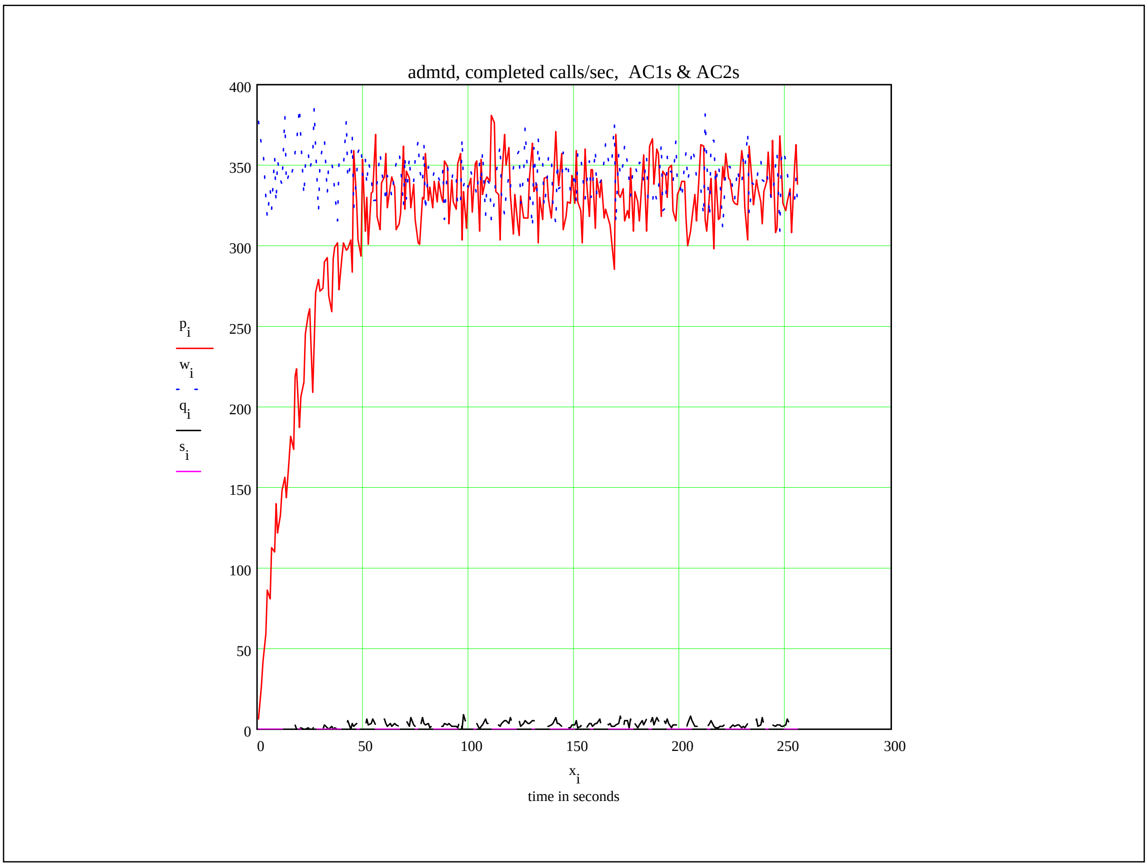
ISUP ACC

Some modelling results

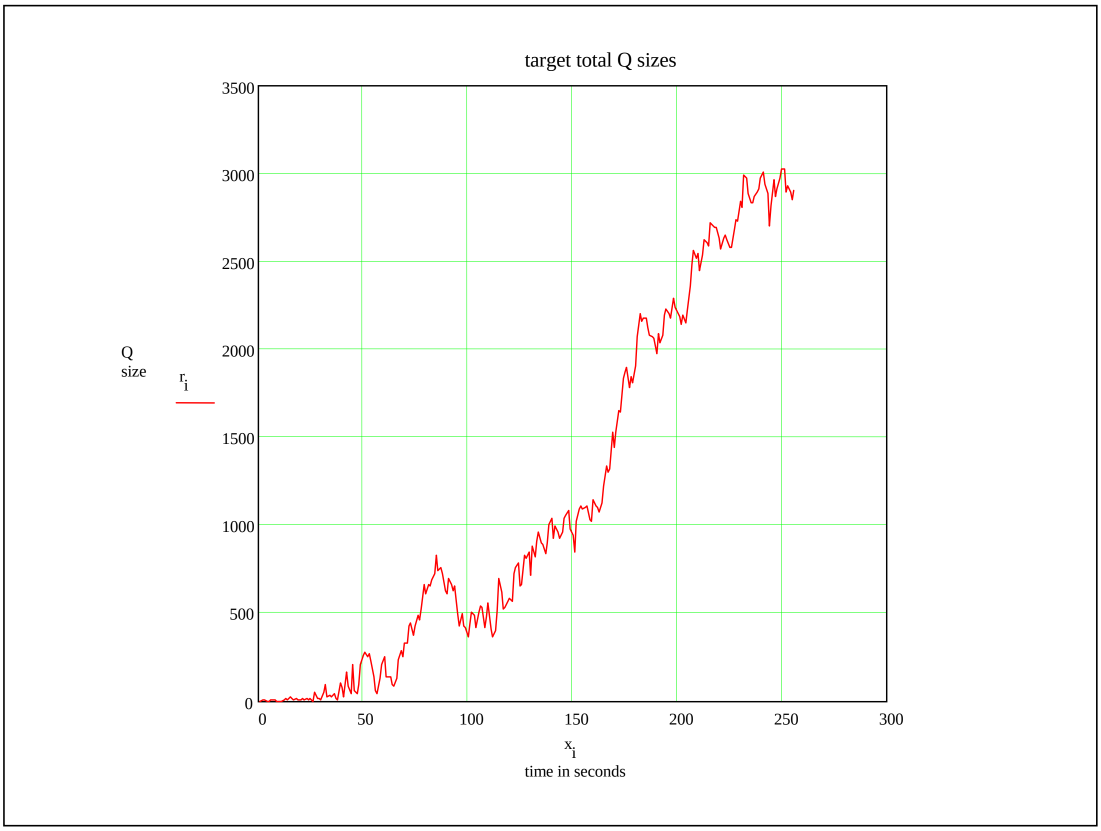


A variable number of sources (in the examples shown here, just 24) supplied originating calls to the target node. The target made use of up to n (usually 8) Network Access Switches (NAS) to connect data paths through from the source switches.
SS No.7 signalling was used from source to target node, with ACC being enabled in the target node.

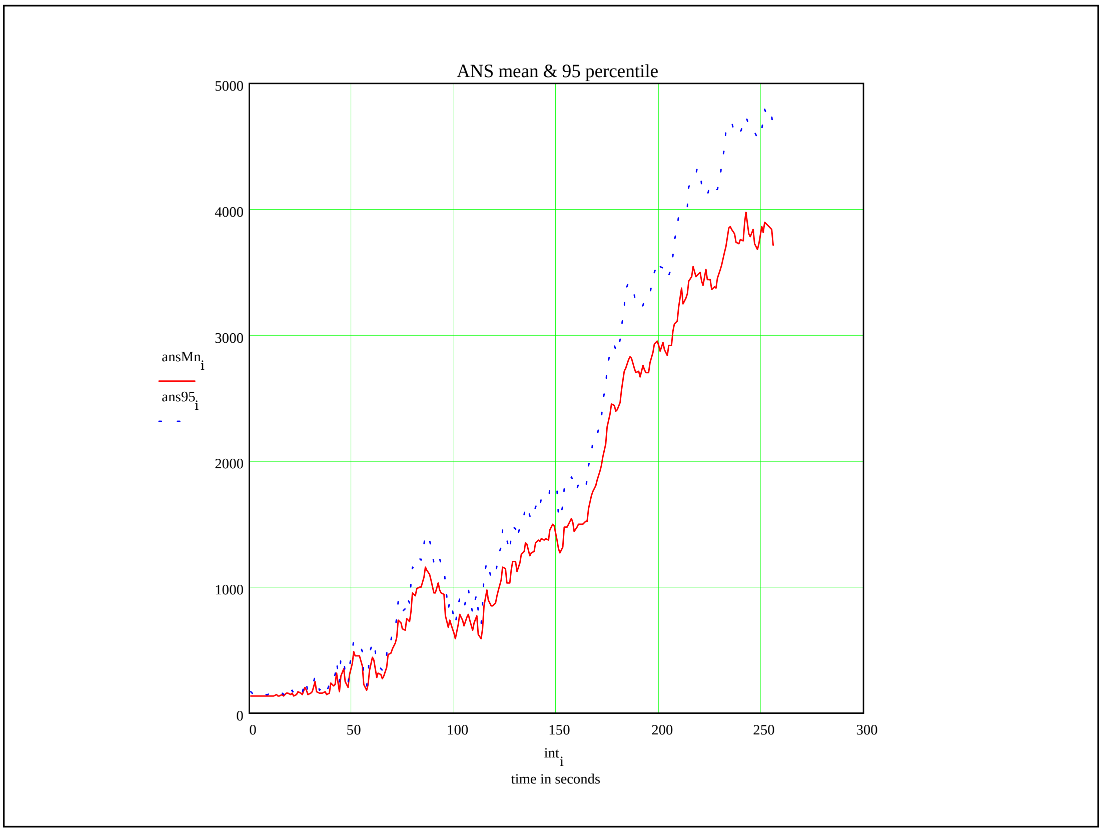
The target was nominally able to accept 300 calls per second without overload.



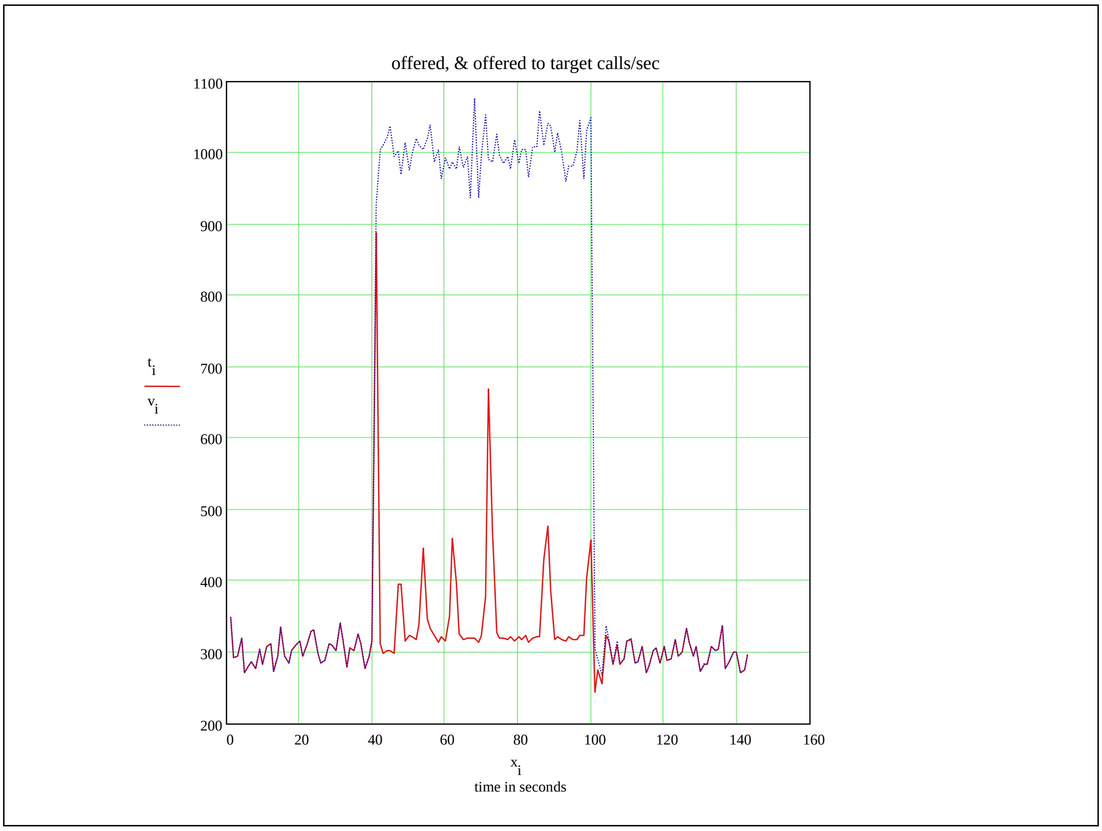
Behaviour when presented with a constant overload (starting from empty target).
350 calls per second offered to target node, call holding time 20 seconds mean, negexp distribution.
Calls admitted by target (top dashed line), calls completed (solid line), RELs received with ACL1 at sources (bottom dashed line), RELs with ACL2 (none seen):
ACC not enabled in sources.



Queues grow without limit : target is sending back REL + ACL for new IAMs, but sources are ignoring its cries for help.

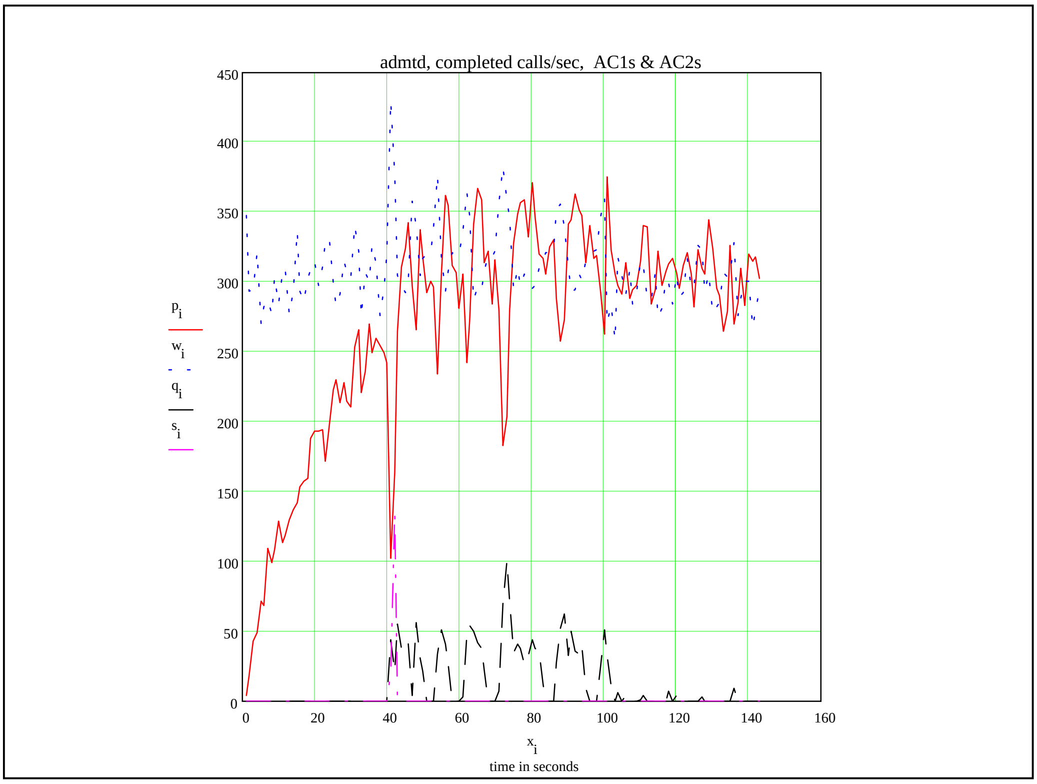


Response time for answer message grows without limit.

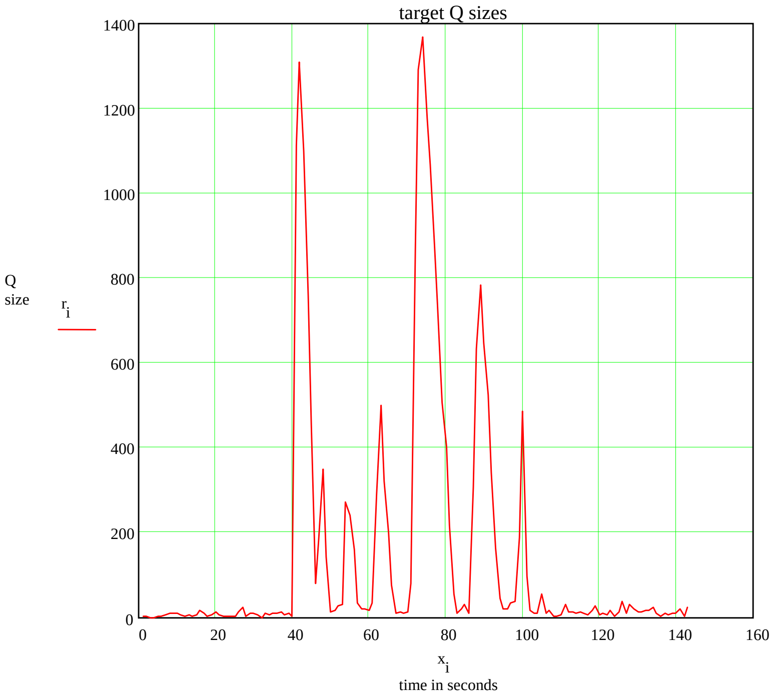


With ACC enabled in sources, and tuned to target’s capacity, large overloads of traffic can be controlled.
In these runs, a burst of traffic is generated evenly from all sources at a total of 1000 calls per second (i.e. 1000/24 calls per second per source) from 40 seconds until 100 seconds, offered to the source traffic limiting mechanism. The number of successful calls, calls offered before the source traffic limiting mechanism, calls offered to the target node after the source traffic limiting mechanism, target node total queue size, ACM, ANS and RLC mean and 95 percentile response times are measured per one second interval.
The ACC traffic limiting mechanism is a leaky bucket scheme per source, with bucket size and leak rate dependent upon the overload level reported to the source by the target. There is a short timer of 300 ms to ignore same-value ACC reports at the source immediately after starting or restarting ACC actions. There is a long (5 second) timer during which ACC actions are continued. If the long timer expires, the congestion level indicated at the source for the target is decremented. If a REL with ACL parameter greater than or equal to the current indicated level is received outside Tshort but within Tlong , timer Tlong is stopped, Tshort is started, which when it expires will cause Tlong to be started for the indicated overload level. If an ACL indicating a higher level of overload is received while Tshort is running, when Tshort expires Tlong is started for the higher indicated overload level. The bucket size per source for indicated overload level 1 is 17, with leak rate 320/24 calls per second per source. For indicated overload level 2, the bucket size is 10, with leak rate 300/24 calls per second per source.

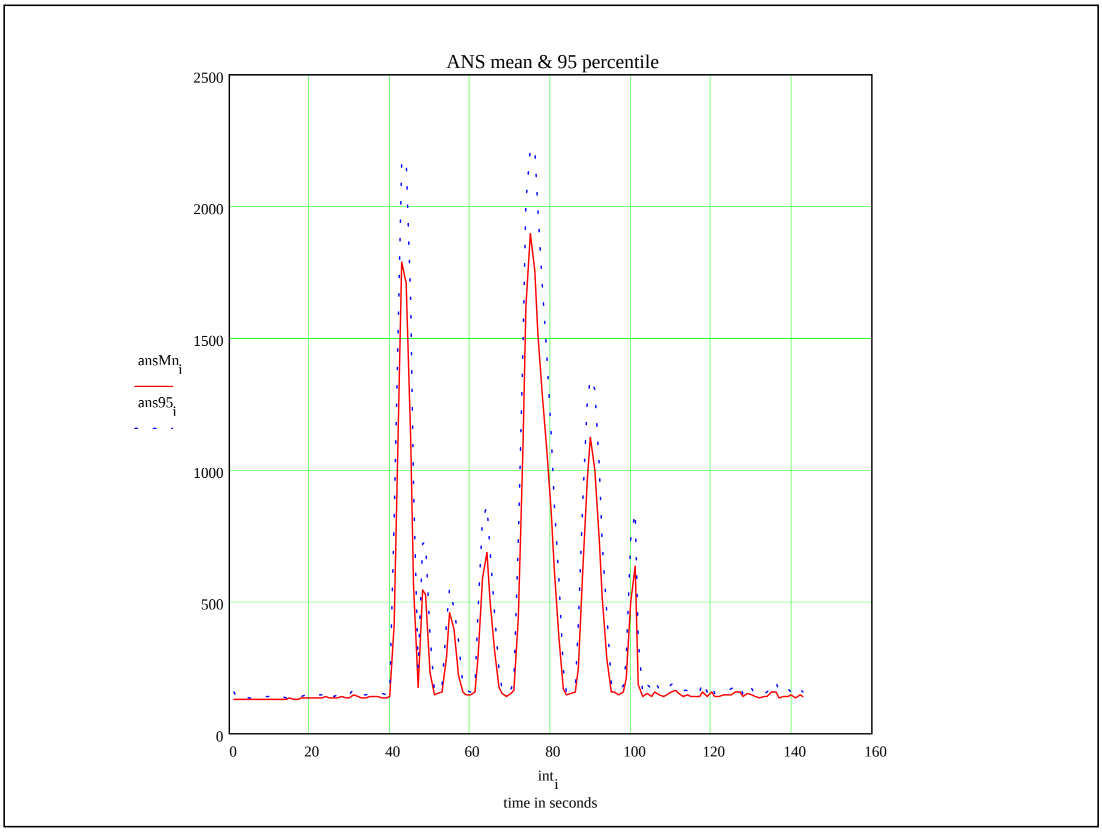
The call holding time was set to a negative exponentially distributed time with mean 20 seconds.



Calls admitted by target (top dashed line), calls completed (solid line), RELs received with ACL1 at sources (bottom dashed line), RELs with ACL2 (spike at 40 secs).



Target total queue size has an initial spike (before the ACC limiting mechanism cuts in at the sources), then a number of following spikes as the source Tlongs expire, which allows them to restart traffic at the burst level, before the ACC mechanism cuts in at the target and then the sources.



Answer response time distribution follows the target total queue size.

Annex F - Siemens Slide Presentation on ISUP ACC

Automatic Congestion Control Performance Analysis

Dr. Gerta Köster
Siemens AG, München

ACC - The Idea

Idea: Protect a node from overload by reducing the traffic at its neighbours.

Necessary steps:

- Determine overload at exchange.
- Transport information to neighboring switches.
- React at neighboring switches according to received information.

Standards: E.412,Q.763,Q.764.

ACC - Siemens Simulation

Goal:

- Get a feeling for how well ACC works in a network.

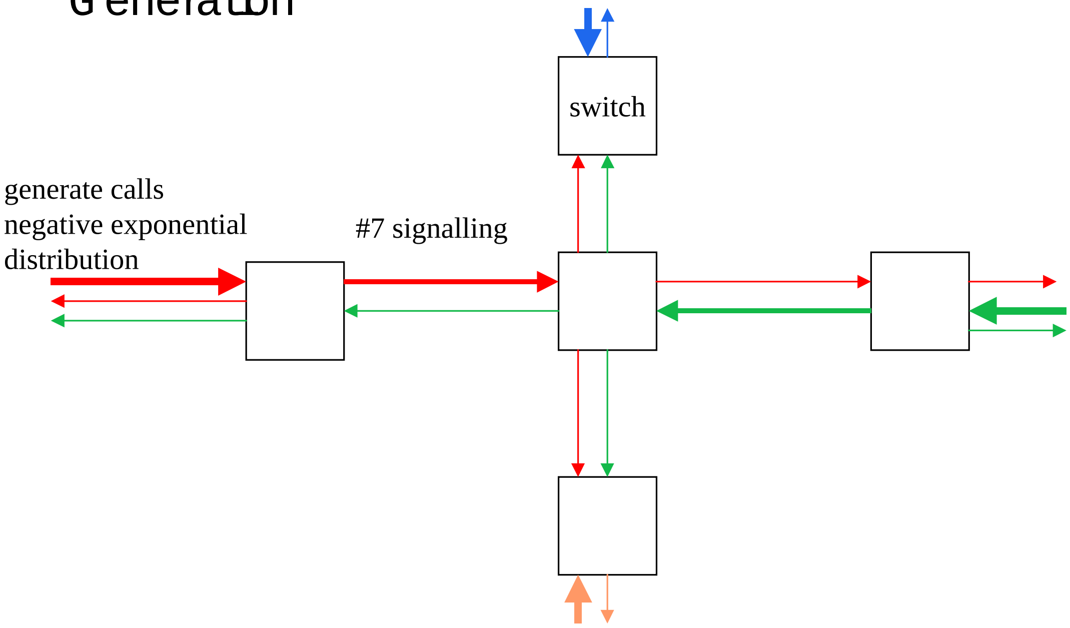
Focus:

- Information of adjacent nodes.
- Reaction upon receipt of information at neighbours.
- Feedback - how does the overloaded switch “react to the reaction“.

Steps:

- Determine and evaluate overload; map to ACC levels.
- Transport of overload information according to standard.
- Reaction of at neighbouring switch: degree of freedom.

ACC - Siemens Simulation: Topology and Traffic Generation

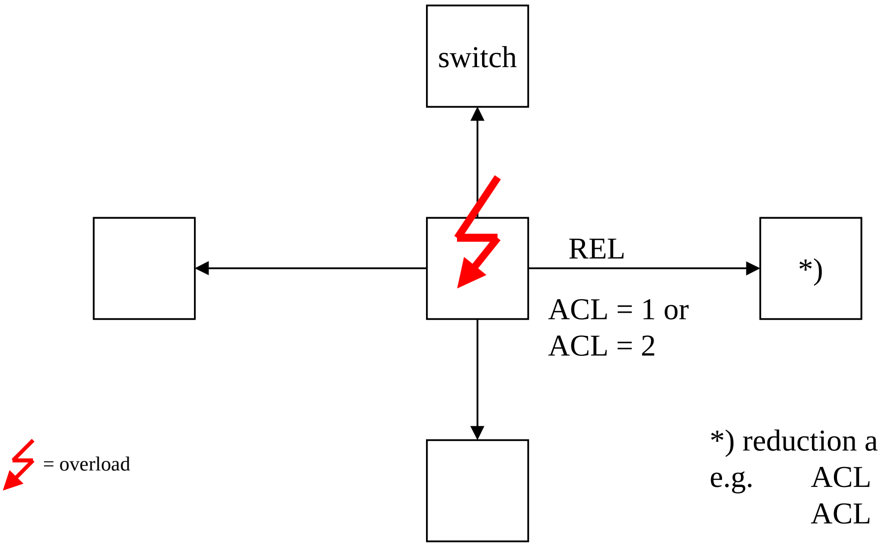


generate calls
negative exponential
distribution

#7 signalling

switch

ACC - Siemens Simulation: Information Transport



*) reduction according to ACL
e.g. ACL = 1: 50%
ACL = 2: 100%

ACC - Simulation: Experiments

Parameters observed during experiments:

- Network throughput - compared to high load.
- CP load - compared to high load.
- Buffer population - compared to high load.

Duration: several minutes.

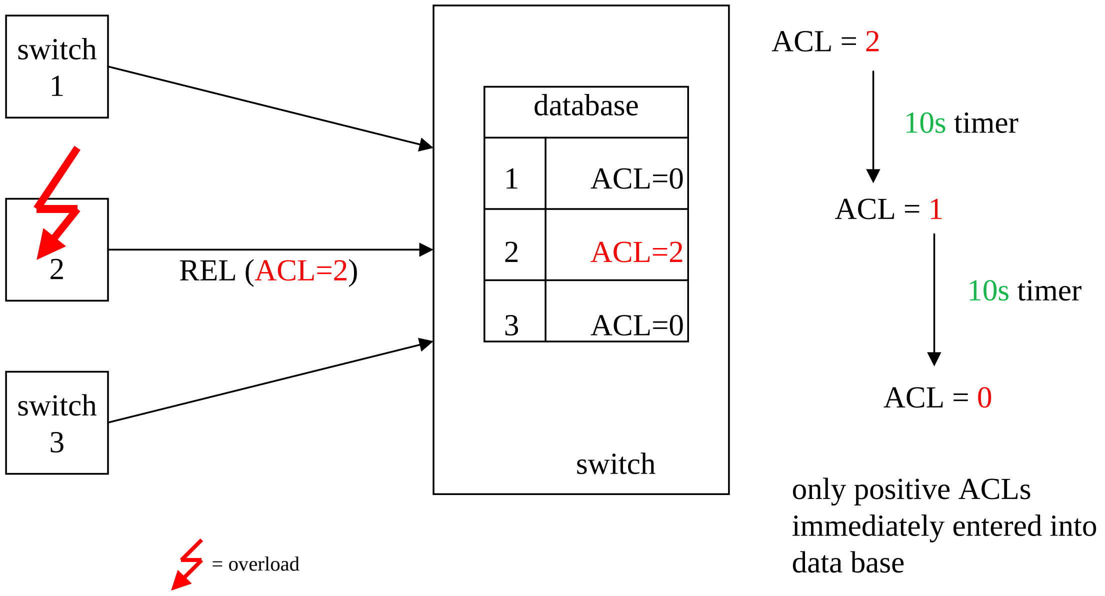
Tests: Comparison with other simulations and measurements.
.

ACC - Experiments

Load situations:

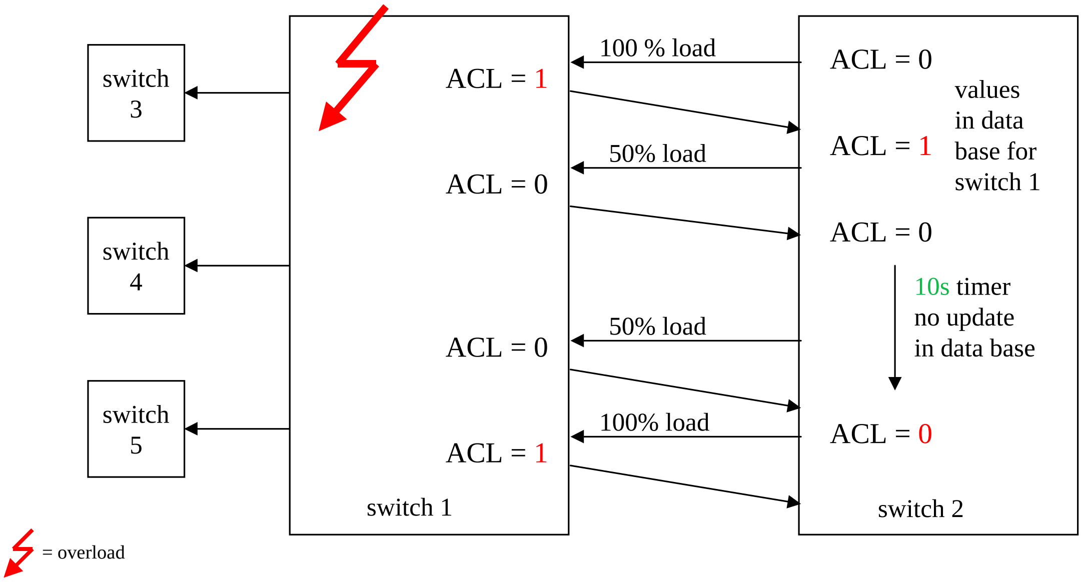
- High load (load B).
- Moderate overload (1.6 x load B).
- Severe overload (2.7 x load B).
- Dramatic overload (5.7 x load B).

ACC Timer Mechanism



only positive ACLs immediately entered into data base

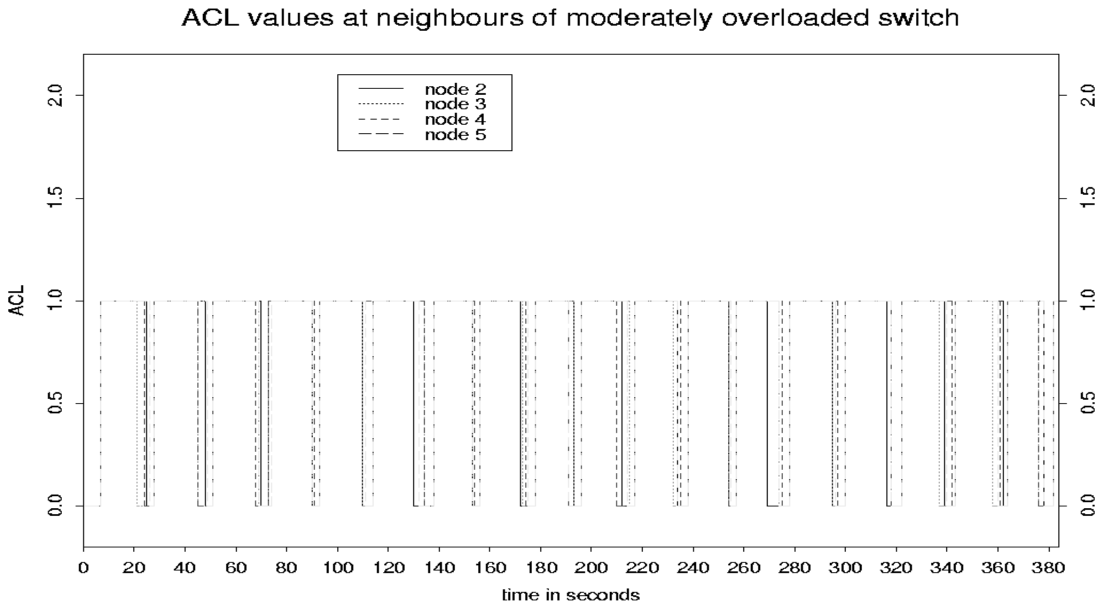
ACC -Bam Door Effect



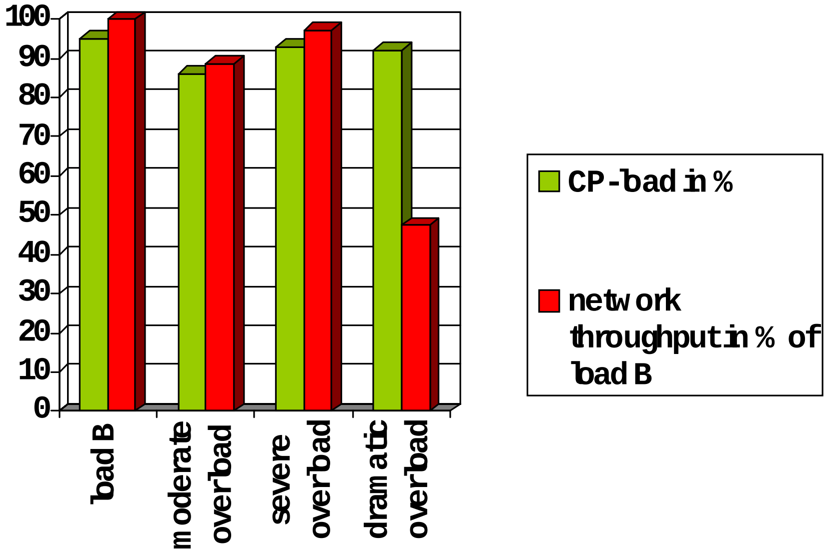
ACC - Further Problems

- Information deficits, when few REL messages are sent.
- Coarse traffic regulation with only 2 values.

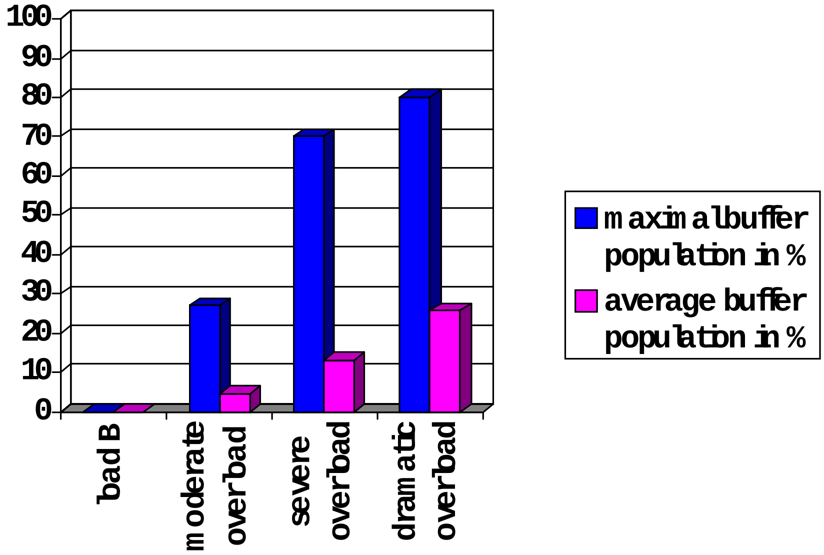
ACC - Simulation Results



ACC - Simulation results



ACC - Simulation results



ACC - Solution strategies

Dead end approaches:

- Adjust timer: A very short timer is equivalent to dispensing with ACC. A long timer further decreases the throughput.
- Choose rejection rates cleverly: Works well in a particular load situation, but not in general.

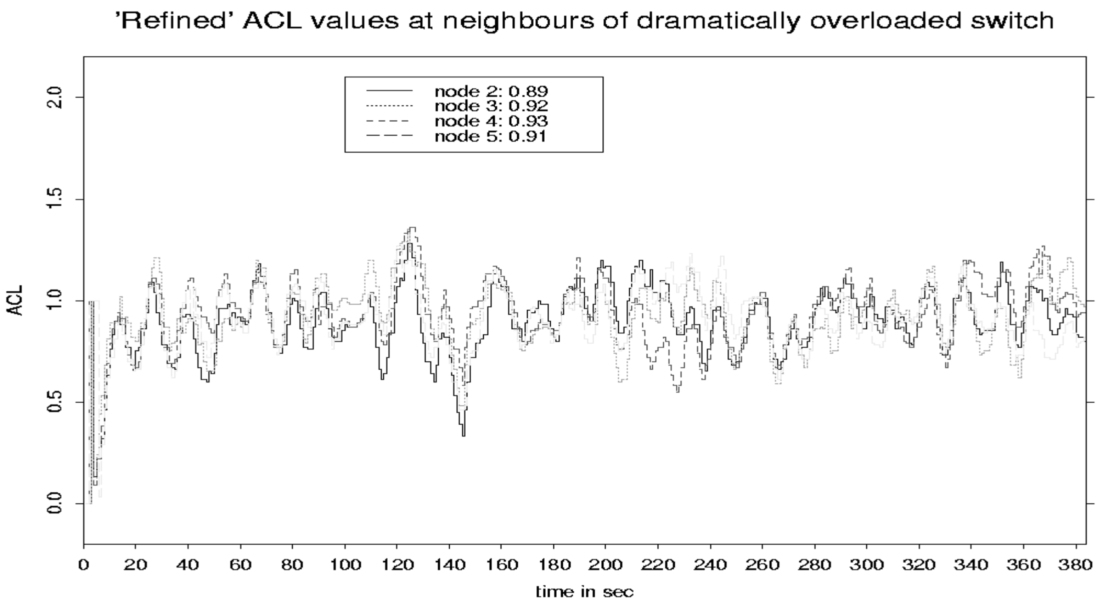
Approach violating the standard:

- Introduce more ACLs.

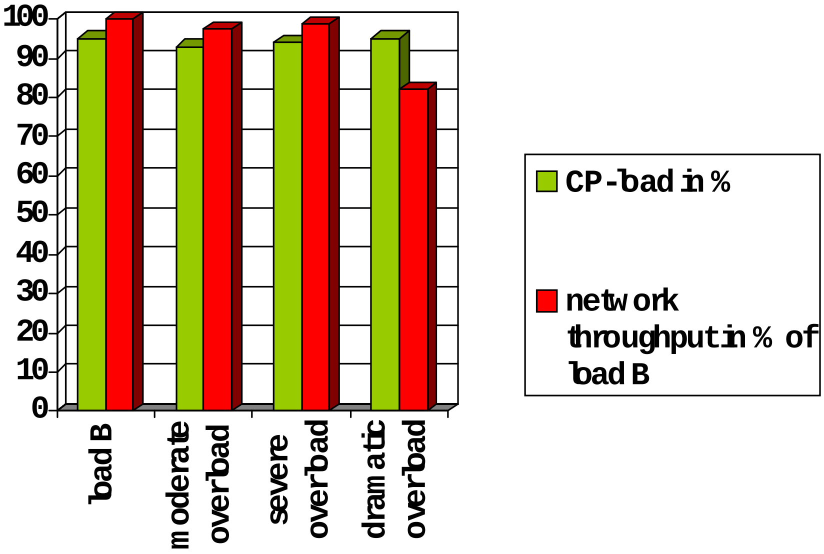
ACC - Solution Strategies

- Retrieve additional information from ACL “history” at the receiving node, that is, compute a refined overload & congestion level (OCL) from the last n (or all) ACLs received in the past.
- Include indirect information contained in “empty” REL messages.
- Map the OCL on 8 reduction levels.
- Smoothen abrupt regulation.

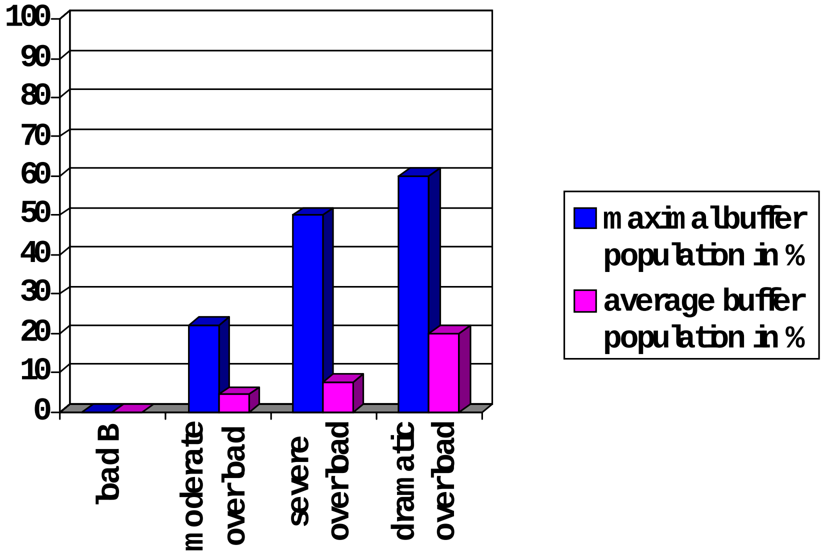
ACC - Simulation Results



ACC - Simulation results for refined ACC



ACC - Simulation results for refined ACC



ACC - Solution Algorithm

International patent pending WO 99/38341, July, 27 1999.

European Patent pending EP 0 932 313 A1, July, 28 1999.