TSG-SA WG3 (Security) meeting #17 **S3-010093**
GÖTEBORG, 27th February – 2nd March 2001

Agenda Item: tbd
Source: Lucent
Title: End-to-End Encryption of Wireless VoIP (E^3 VoIP)

Document for: Discussion and Decision

1 Scope

2 Introduction

End-to-End Encryption (E^3 VoIP) is recommended for Wireless VoIP and Circuit-Switched Voice because it increases security and performance and reduces system complexity. Although this contribution focuses on End-to-End Encryption (E^3 VoIP) for Wireless VoIP, the methods herein are equally applicable to Circuit-Switched Voice. In fact, applicability extends to CS-PS connections and more generally to any termination, for example, an IP Phone, that contains an AMR coder

Two methods of E^3 VoIP are examined and the non-synchronized one is recommended. A request for feedback on the feasibility of implementing E^3 VoIP in the R5 time frame is being liased from S1 to S3.

3 Advantages of E^3 VoIP

3.1 Increased security as perceived by the mobile owner

The mobile owner may not differentiate between the public Internet and VPN internets that bridge Carriers' proprietary domains. Thus the owner may (erroneously) associate the somewhat publicised security vulnerability of the public Internet with his or her own calls. Here, end-to-end encryption can provide the mobile owner with a feeling of security.

3.2 Increased security to the Carrier

With end-to-end encryption of media, a Carrier could interoperate with IP backbone providers in a manner where total trust in the provider's security is not needed. For example, the backbone provider may not have encryption implemented at the time that service is first provided to the Carrier. End-to-end encryption would avoid this issue.

3.3 Network Security is increased overall

Security is like a chain in that it is only as strong as its weakest link. If media were link encrypted instead of end-to-end encrypted, there would be more opportunities for attack. Such an attack need not recover keys or keystream, but could simply take advantage of the fact that the content is unencrypted at the junctions between links. With end-to-end encryption, the content is encrypted throughout.

When the Air Interface is separately encrypted, handovers present challenges. Session keys must be quickly sent around the network. There is a risk of non-delivery of a needed key due to bandwidth or authentication center limitations. Moreover, if the new system is not set up with compatible security, the session may revert to no encryption. End-to-end encryption is transparent to handovers.

3.4 Complexity is reduced

Replacing the encryption of several links with a single encryption means much simpler key management.

Another advantage occurs at handovers. Currently, when the Air Interface is encrypted, session keys must be sent either within a system, or between systems. With end-to-end encryption this overhead disappears and no encryption-related events need to occur at handover.

3.5 Performance is enhanced

Replacing the encryption of several links with a single encryption could lower link setup times.

At handover, VoIP call quality could be enhanced because there would be no waiting time at the new BTS for reception of the session key and regeneration of key schedules.

4 Key Management and Lawful Surveillance

The IP Multimedia Subsystem would be responsible for the key management procedures required to support the end-to-end encryption of voice calls controlled by the IP Multimedia Subsystem. Consequently, the IP Multimedia Subsystem would have the key information required to potentially support lawful surveillance requirements for these types of calls.

One such possible key management implementation would be to randomly generate a session key within the system and send it securely to both ends.

5 End-to-End Encryption Summary

When compared with link encryption, E^3 VoIP provides several advantages: security increase, complexity reduction, and performance enhancement. The need for a newly defined key management system and a new Law Enforcement issue do not seem to be unduly problematic.

6 Implementation

Here we examine two end-to-end encryption methods for VoIP: SE^3 (Synchronized End-to-End Encryption) and NSE^3 (Non-Synchronized End-to-End Encryption). The former is a standard encryption method that uses cryptosync.

6.1 Definition of Cryptosync

Cryptosync (cryptosynchronization) is one method of allowing the state of a cipher to change with each frame by synchronizing states at both ends. NSE^3 is another method within the context of this document. More generally, there are three common methods of encryption used in Wireless:

1. Stream encryption
2. Block encryption
3. Fixed mask

Stream encryption uses cryptosync to produce a stream of random bits called keystream, which is generally XORed with the plaintext to produce ciphertext.

Block Encryption transforms a block of plaintext into ciphertext and uses the entropy of the plaintext to provide unique encryptions. If the entropy is sufficient, block encryption need not use cryptosync.

A fixed mask XORs the same set of secret bits to each message of plaintext. Since the mask is fixed, no cryptosync is needed. However, unless the plaintext is unknown and has negligible redundancy, this method is insecure.

Generally, stream encryption is the preferred method in Wireless because of its high security, ease of implementation, and lack of error extension. However regarding VoIP encryption, if the two ends are not perfectly synchronized, voice is effectively muted.

6.2 *Difficulties with Cryptosync for E^3 VoIP*

During the body of a call, end-to-end encryption frees the system of security-related tasks such as transferring session keys at handover. Unfortunately, cryptosync would mitigate this freedom. It would re-involve the system with tasks such as maintenance of synchronization at handover and ensuring synchronization when dropping or adding of frames to deal with asynchronous frame clocks at the two ends.

Even with system involvement, cryptosync is problematic to maintain, especially after a handover. A possible impact here is performance degradation. Also, in the network, cryptosync adds complexity and requires bandwidth.

One proposed method of sending VoIP over the network would use the RTP protocol (Real Time Protocol). RTP would ensure that voice frames did not arrive out of order. If cryptosync were to be implemented, it then would likely try to leverage off this protocol perhaps by using the sequence number field. However, the Air Interface has insufficient bandwidth to carry this sequence number and so the number must be regenerated in the mobile and BTS. RTP is currently defined to not increment its sequence number for silence frames, thus either the RTP sequence number would be redefined to handle encrypted speech, or the timestamp field is used instead, assuming that it is free. Either of these changes may violate other system requirements.

Even if RTP regeneration were enabled by either of the above, two further difficulties would remain:

- Conveying cryptosync in a handoff, particularly in an inter-system handoff
- Dealing with cycle slips between near-end and far-end clocks

Conveying cryptosync in a handoff would mute voice for a period of time and would require higher-layer synchronous messaging capability as shown by the following example implementation:

Reverse Direction

- Silence frames are initially sent from network entity that contains RTP terminus. This will interrupt speech.
- Mobile sends BTS a message containing a sequence number and frame number
- BTS sends an “ACK” back to mobile
- BTS then sends a synchronous message containing sequence number to network entity that contains RTP terminus.
- RTP is resynced and voice transmission commences in reverse direction

Forward Direction

Similar to above ...

In conclusion, implementing cryptosync for SE^3 would have the following known impact:

- Redefinition of RTP, or using timestamp field (if free)

- System-level complexity due to multi-entity, synchronous and fast asynchronous messaging needed to deal with handoffs and cycle slips
- Probable voice degradation due to muting until cryptosync is established and re-established
- Irrecoverable voice muting if cryptosync is off by even one count
- Additional network bandwidth needed for cryptosync

7 NSE^3 (Non-Synchronized End-to-End Encryption)

The entropy in voice allows another form of end-to-end encryption, which we term NSE^3 . This approach does not use cryptosync and thus makes it easy for the network to be transparent to E^3 VoIP. An architectural overview is shown below:

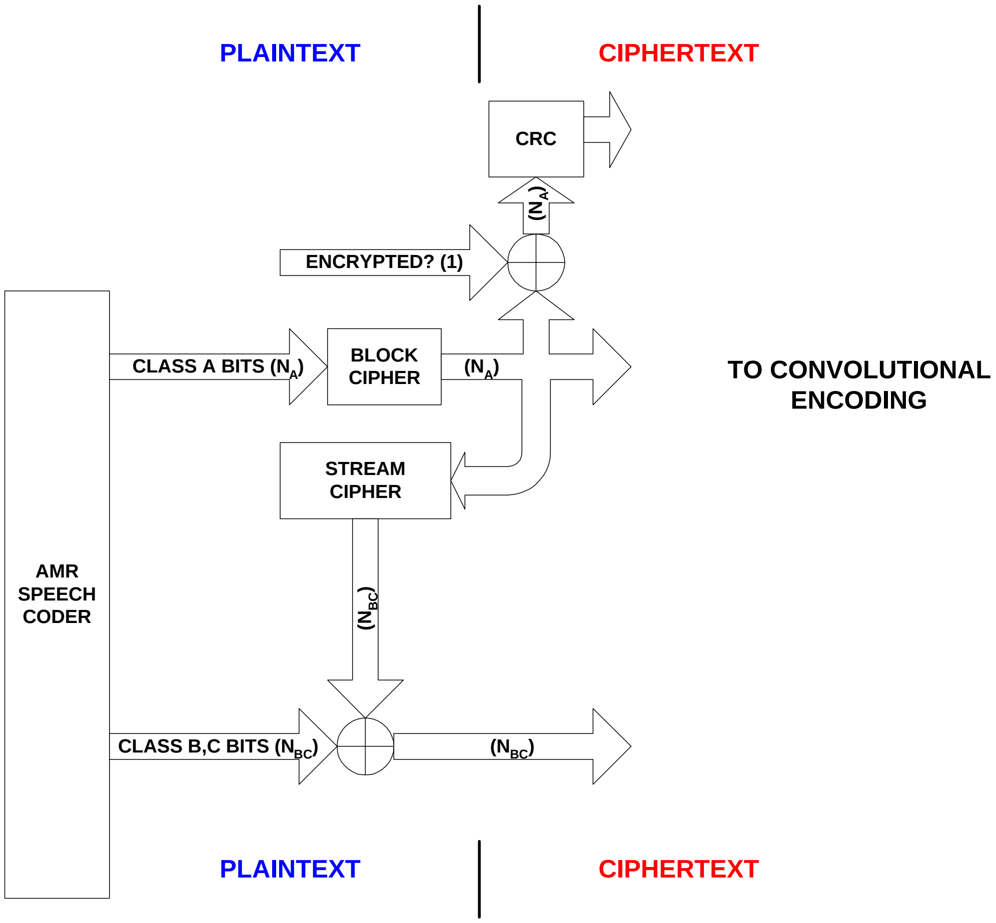


Figure 1 - NSE^3 Architecture

The AMR speech coder's output is partitioned into 3 classes of bits prior to channel coding. In UMTS, these bits are called Class A, B, and C. These bits closely, but not exactly, correspond to the Class 1a, 1b, and 2 bits in GERAN:

- Class A (Class 1a in GERAN): Protected by a checksum and not used if the checksum fails. Also protected by strong convolutional coding. These are the most perceptually important bits.
- Class B (Class 1b in GERAN): Not protected by a checksum, and may or may not be used if Class A checksum fails. Also protected by strong convolutional coding.
- Class C (Class 2 in GERAN): Not protected by a checksum, but used even if Class A checksum fails. Protected by weaker convolutional coding in UMTS and not protected in GERAN. However, there is little or no perceptual degradation due to their use under checksum fail.

The selective use of bits under a checksum fail is termed “Bad Frame Masking”. Not described here is how “Bad Frame Masking” extrapolates Class A and some Class B bits from prior “good” frames.

The above partitioning allows a synergy between “Bad Frame Masking” and encryption. The Class A bits are encrypted with a block cipher and the Class B and C bits are encrypted by a stream cipher driven by the output of the block cipher. It is the entropy of the Class A bits that provides unique encryptions from frame to frame. The synergy is explained by considering two cases: CRC Pass, and CRC Fail.

CRC Pass

Class A bits are generally unerrored and decrypted correctly (except for rare instances of a false-positive CRC). Since Class A ciphertext is thus unerrored, the stream cipher’s keystream output matches at both ends and thus no errors are added to Class B and Class C bits. In other words, encryption is transparent.

CRC Fail

Class A and in some cases part of the Class B bits are not used from the current frame so it does not matter how their respective decryptors are performing.

Some Class B and Class C bits are passed but totally errored by the decryption, and thus comprise random noise. However, A/B and MOS testing shows that replacing these bits by noise is essentially not perceptible under a CRC-fail condition. This is because the frame was errored enough to have yielded Class A errors after strong error correction and the additional errors on the less important bits do not make a perceptually significant difference.

In other words, encryption is again transparent.

7.1 Another Potential Advantage of NSE^3

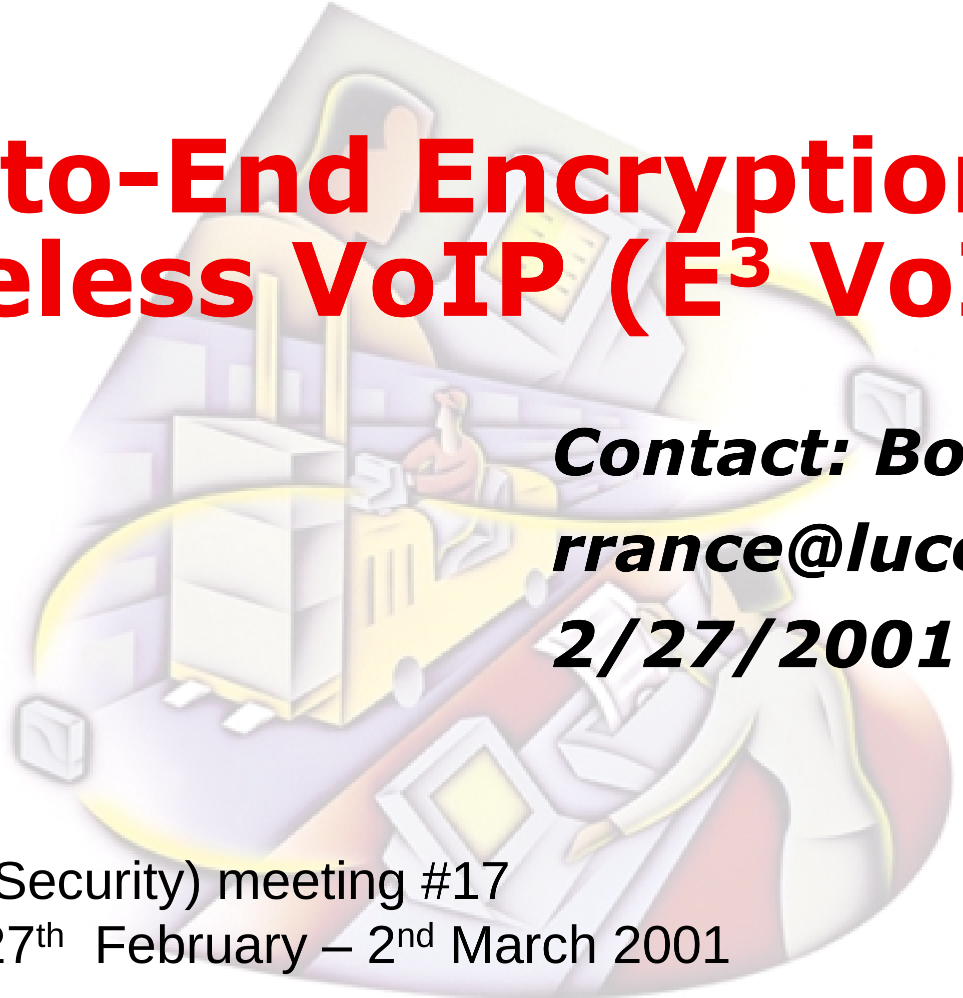
Currently, there are no plans to provide a TFO (Transcoder-Free Operation) connection between a circuit-based mobile and a VoIP-equipped mobile or gateway. This would be a connection where the AMR coders in each entity communicated directly. However, if such were ever implemented, NSE^3 would facilitate end-to-end encryption. Alternatively, without NSE^3 in this hypothetical application, initializing and maintaining cryptosync across the CS-PS boundary would add another layer of difficulty. In conclusion, NSE^3 would allow a universal end-to-end encryption approach that would transparently weather the transition from circuit-based mobile voice to IP-based mobile voice.

In addition, NSE^3 would easily allow a connection from any of the aforementioned termini to a generic IP phone. The only requirement is that the connection is TFO, which in fact is a requirement anyway for end-to-end encryption.

8 References

[1] 3G.IP Requirements Document, Version 1.0, section 3.7.1.1.2, "Voice & Data Privacy". May 5, 1999.

End-to-End Encryption of Wireless VoIP (E^3 VoIP)



Contact: Bob Rance

rrance@lucent.com

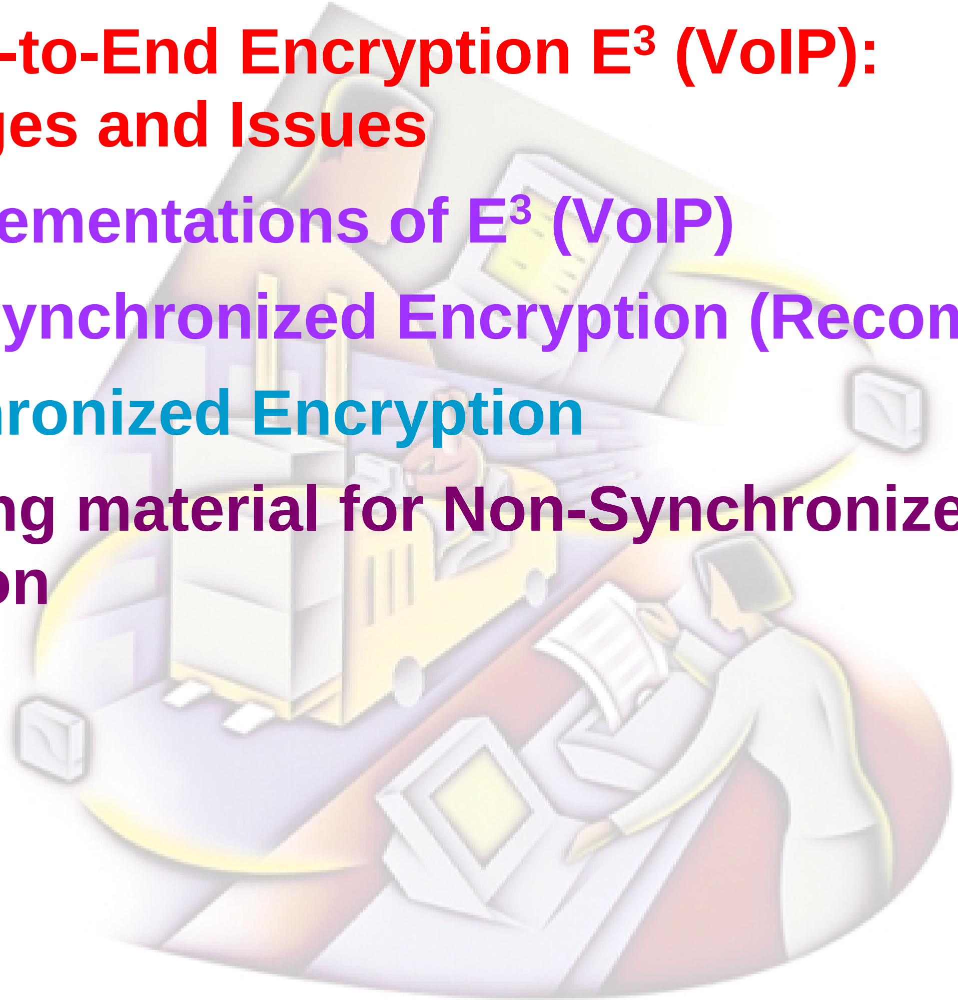
2/27/2001

TSG-SA WG3 (Security) meeting #17
GÖTEBORG, 27th February – 2nd March 2001

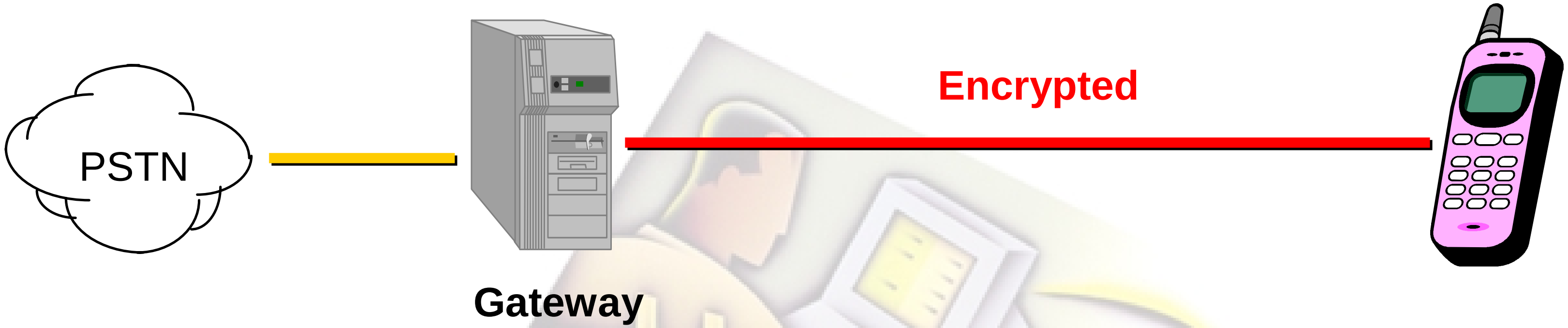
S3-0100XXX

Contents

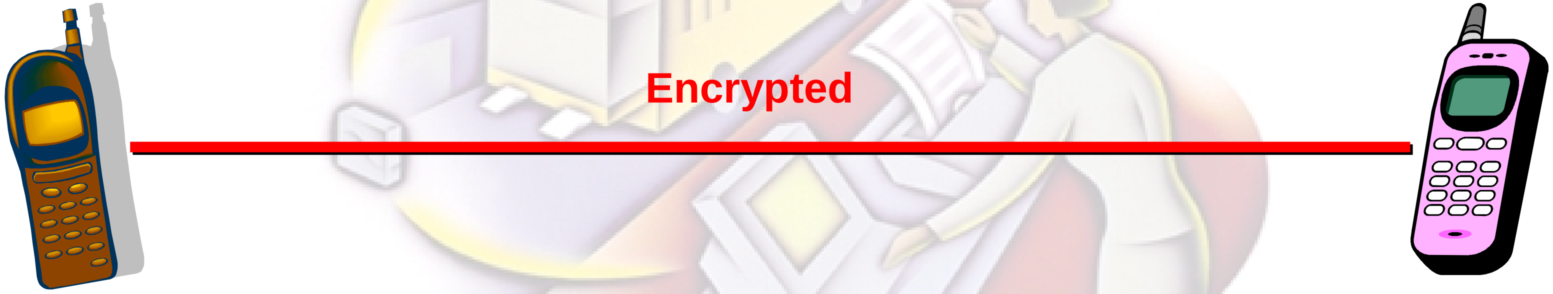
- VoIP End-to-End Encryption E^3 (VoIP): Advantages and Issues
- Two Implementations of E^3 (VoIP)
  - Non-Synchronized Encryption (Recommended)
  - Synchronized Encryption
- Supporting material for Non-Synchronized Encryption



E^3 VoIP Architectures



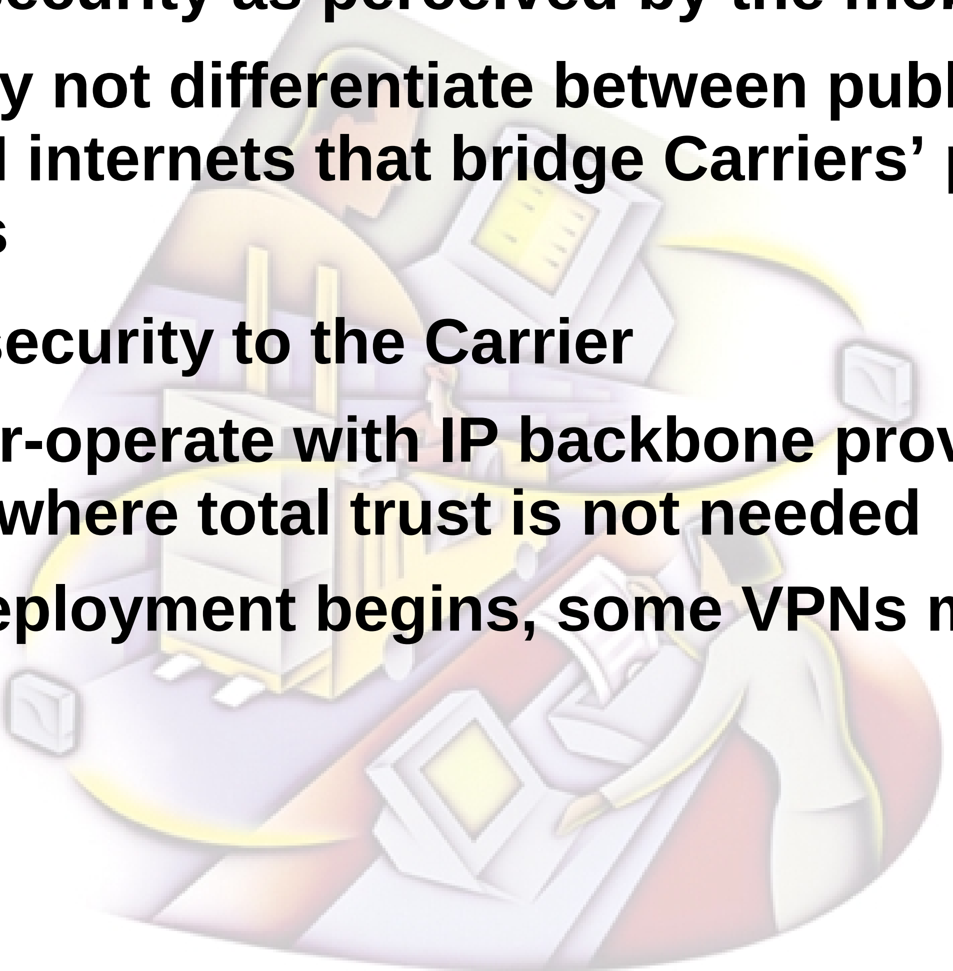
Gateway to Mobile



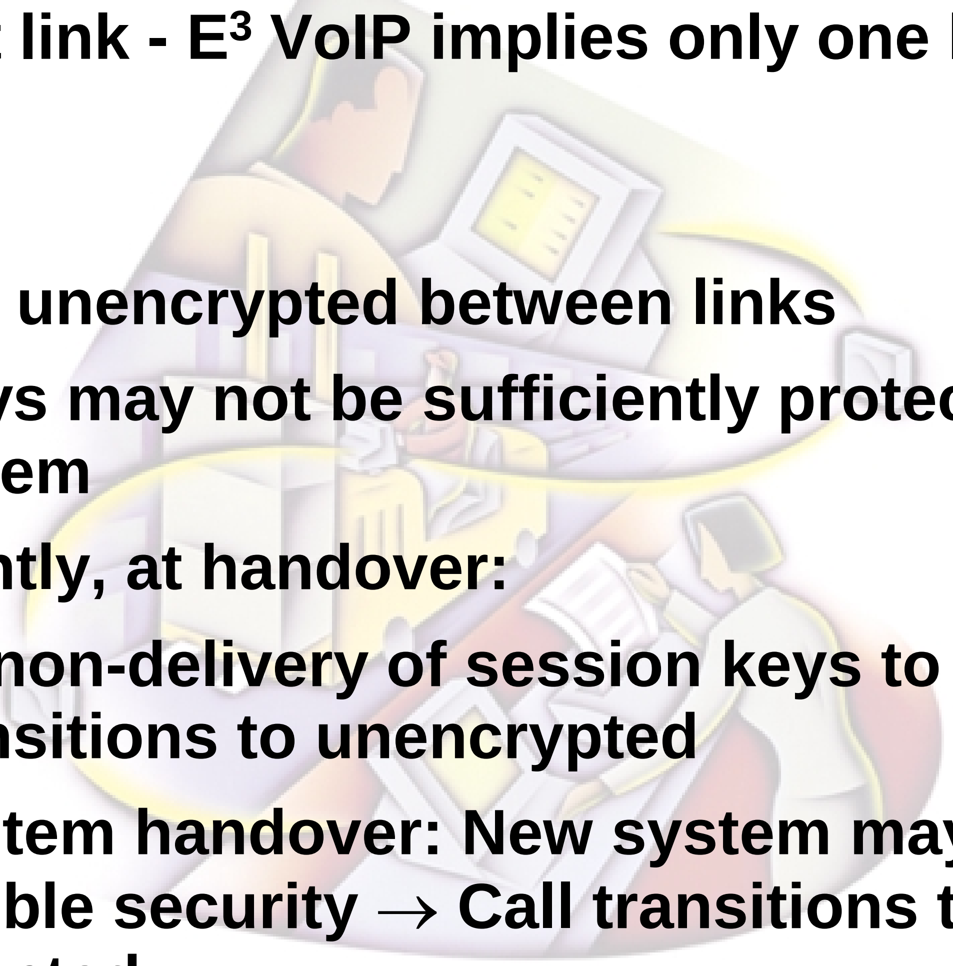
Mobile to Mobile

Security Advantages of E^3 VoIP

- Increased security as perceived by the mobile owner
  - User may not differentiate between public Internet and VPN internets that bridge Carriers’ proprietary domains
- Increased security to the Carrier
  - Can inter-operate with IP backbone providers in a manner where total trust is not needed
  - When deployment begins, some VPNs may not be secured



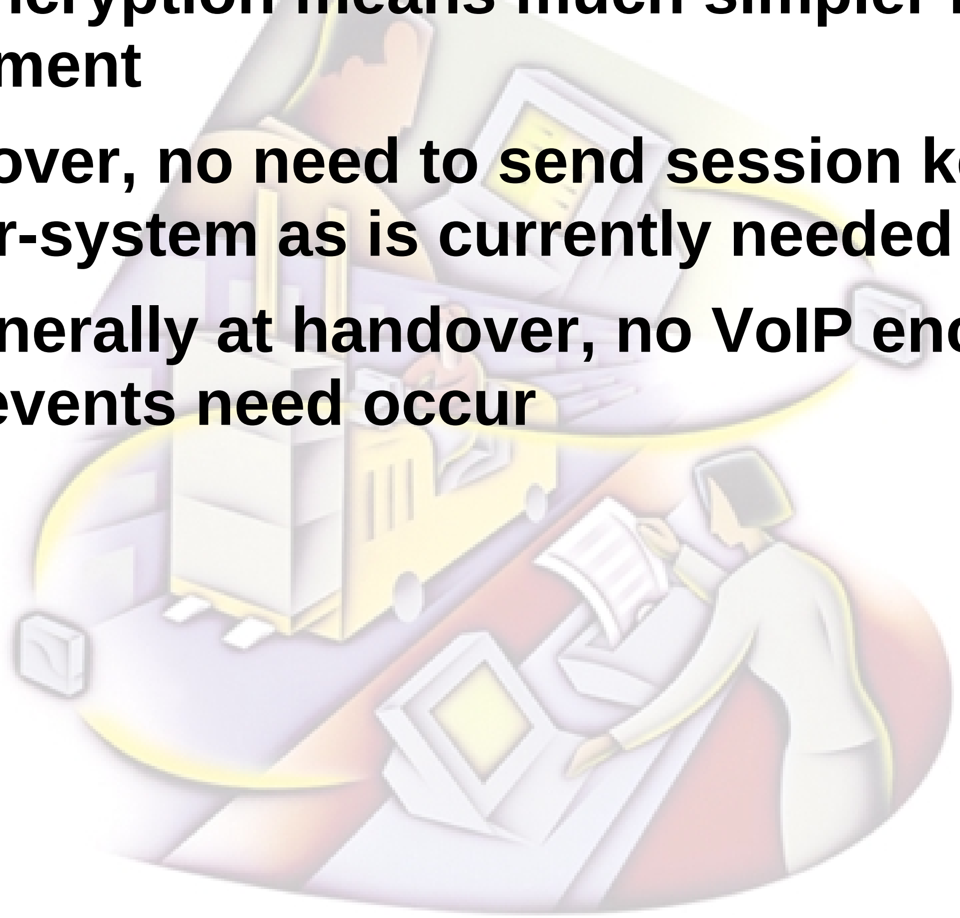
Network security increased overall



- Security is like a chain in that it is only as strong as its weakest link - E^3 VoIP implies only one link to be secured
- Currently:
  - Voice is unencrypted between links
  - Link keys may not be sufficiently protected within the system
- Also currently, at handover:
  - Risk of non-delivery of session keys to mobile → Call transitions to unencrypted
  - Intersystem handover: New system may not have compatible security → Call transitions to unencrypted

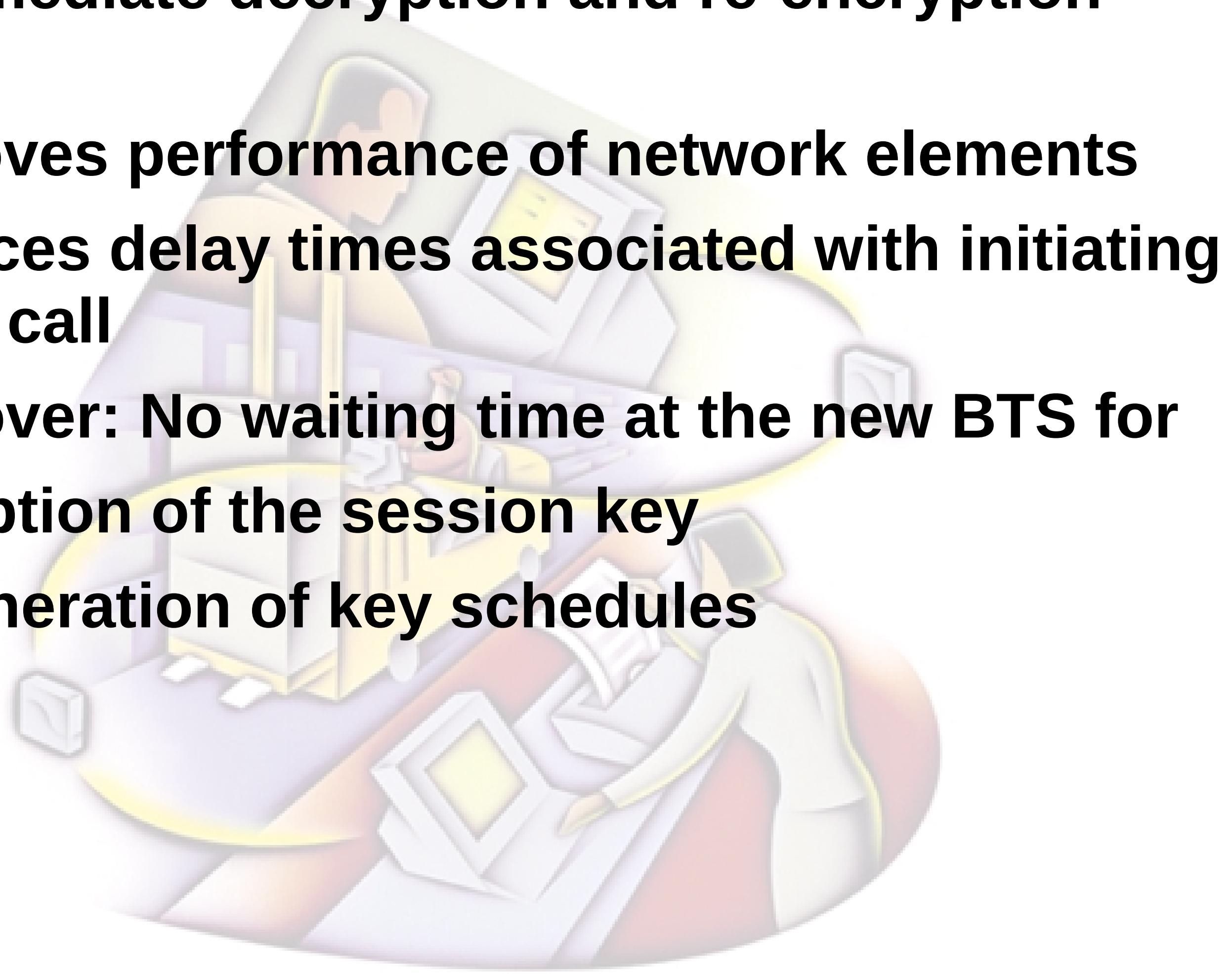
System Complexity is Reduced

- Replacing the encryption of several links with a single encryption means much simpler key management
- At handover, no need to send session keys intra and inter-system as is currently needed
- More generally at handover, no VoIP encryption-related events need occur



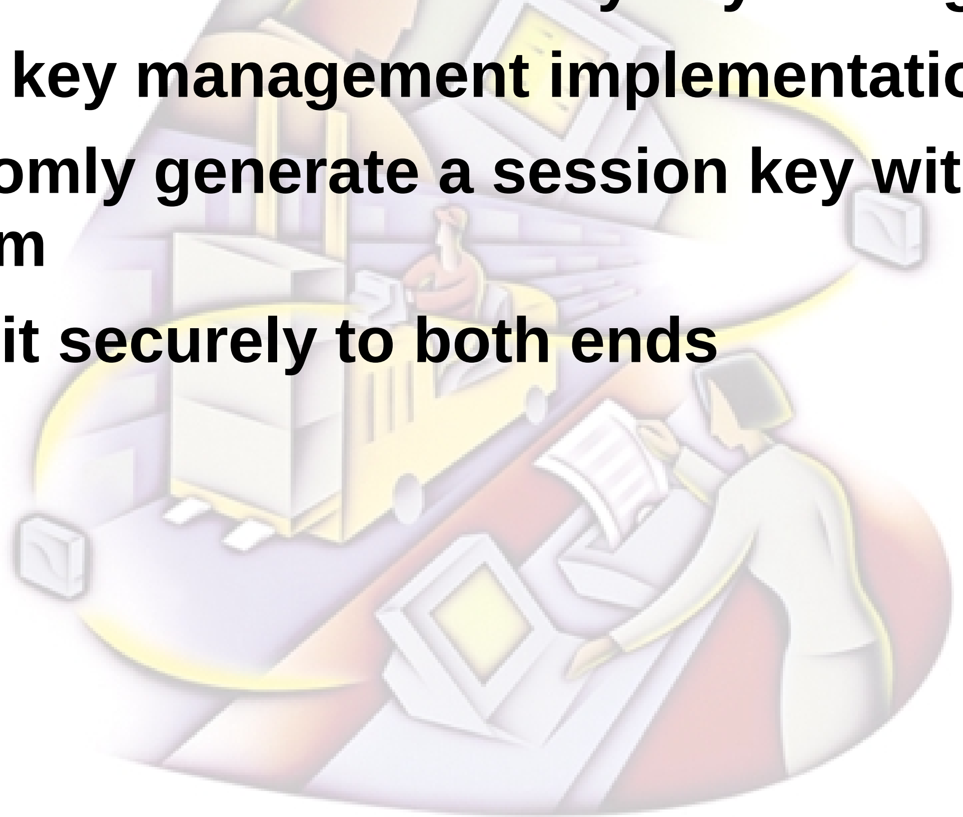
Performance is Enhanced

- No intermediate decryption and re-encryption required
  - Improves performance of network elements
  - Reduces delay times associated with initiating voice call
- At handover: No waiting time at the new BTS for
  - Reception of the session key
  - Regeneration of key schedules



Lawful Surveillance and Key Management

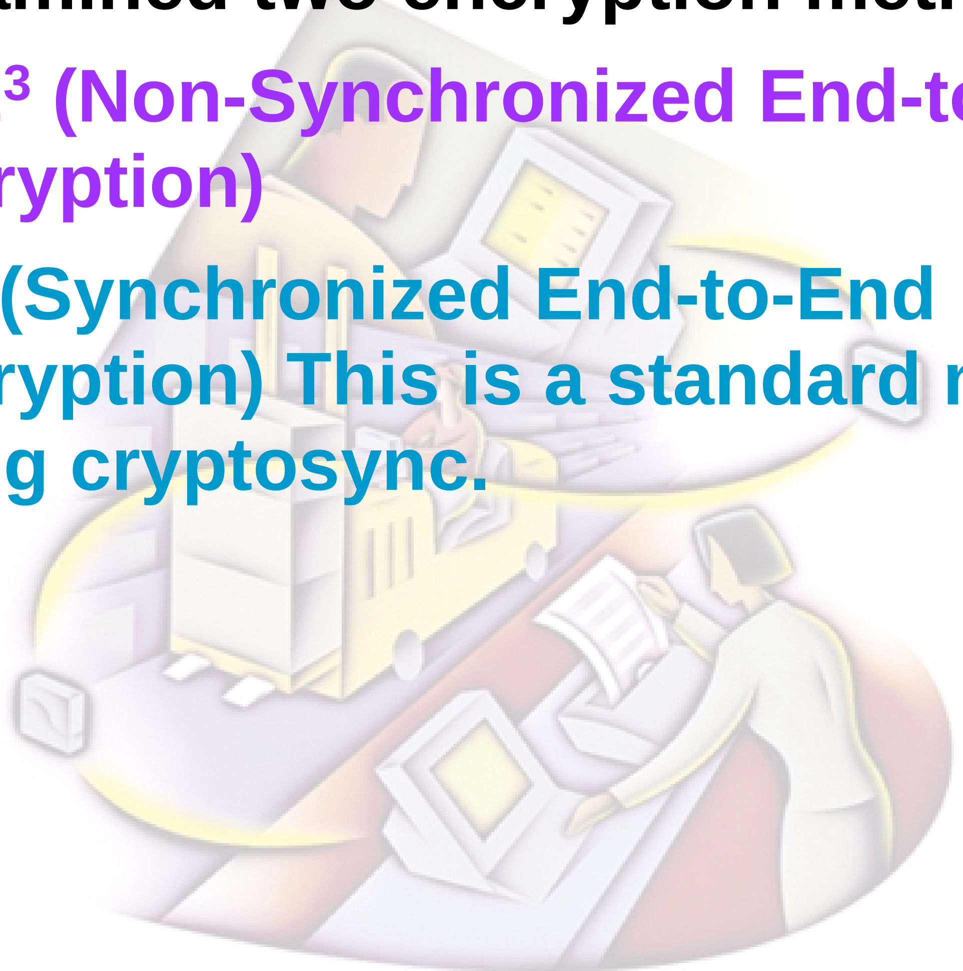
- Since clear speech does not exist within network, another Lawful Surveillance mechanism must be defined: Government-friendly key management
- Possible key management implementation
  - Randomly generate a session key within the system
  - Send it securely to both ends



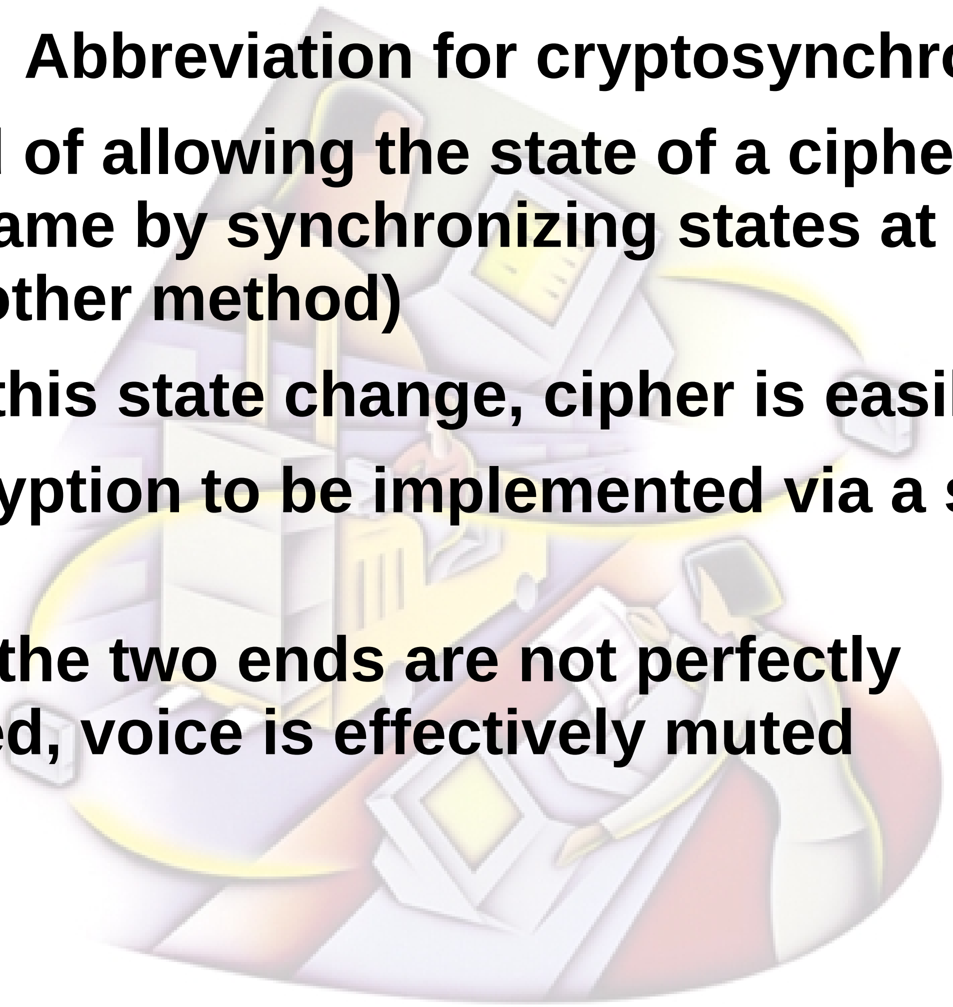
Implementation

Two Methods

- We examined two encryption methods:
  - NSE^3 (Non-Synchronized End-to-End Encryption)
  - SE^3 (Synchronized End-to-End Encryption) This is a standard method using cryptosync.



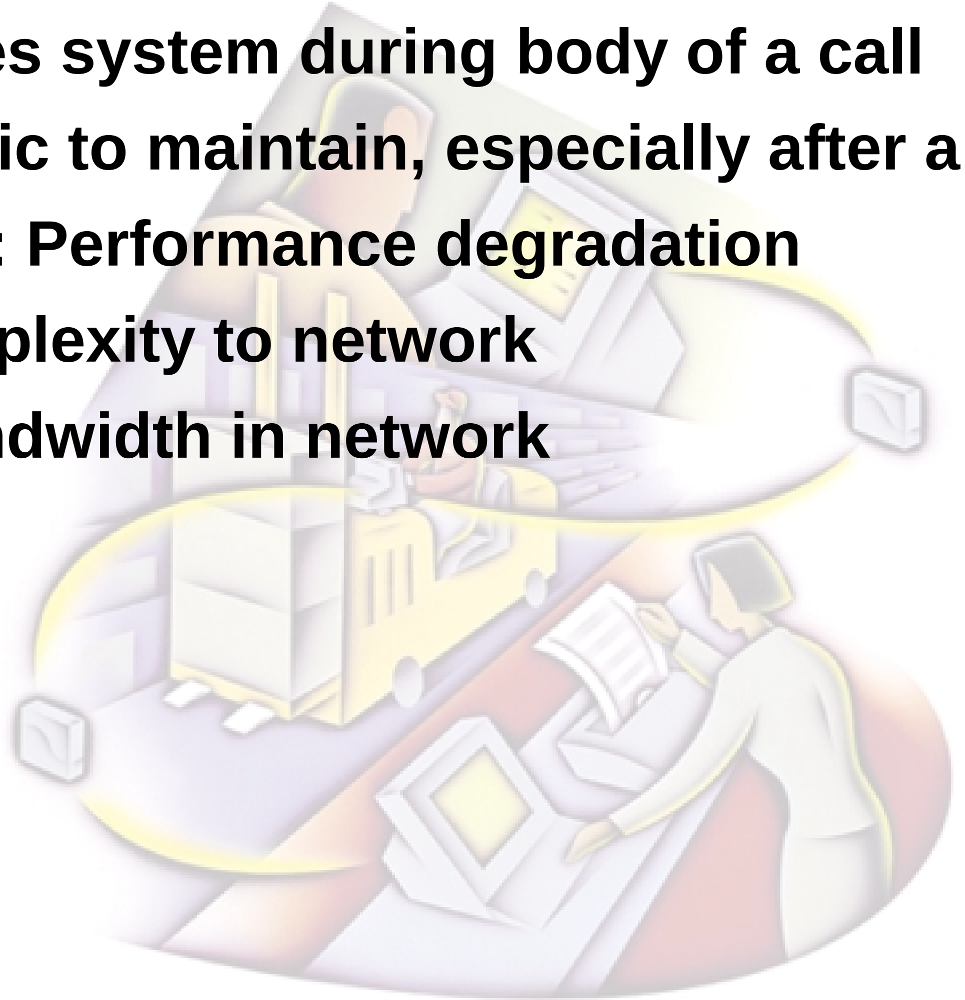
What is Cryptosync?



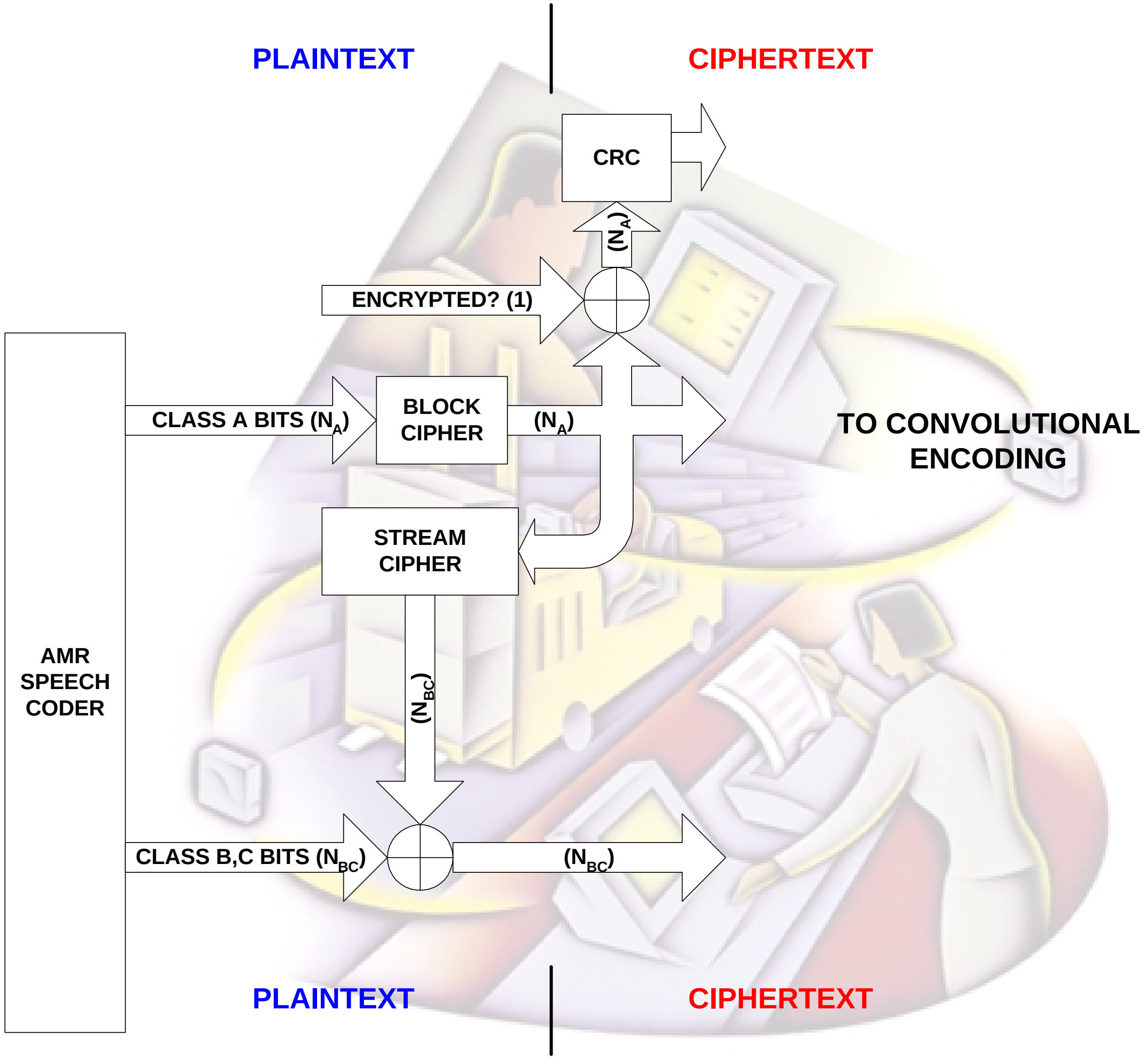
- Cryptosync: Abbreviation for cryptosynchronization
- One method of allowing the state of a cipher to change with each frame by synchronizing states at both ends (NSE^3 is another method)
  - Without this state change, cipher is easily broken
- Allows encryption to be implemented via a stream cipher
- *However*: If the two ends are not perfectly synchronized, voice is effectively muted

Difficulties with Cryptosync for E^3 VoIP

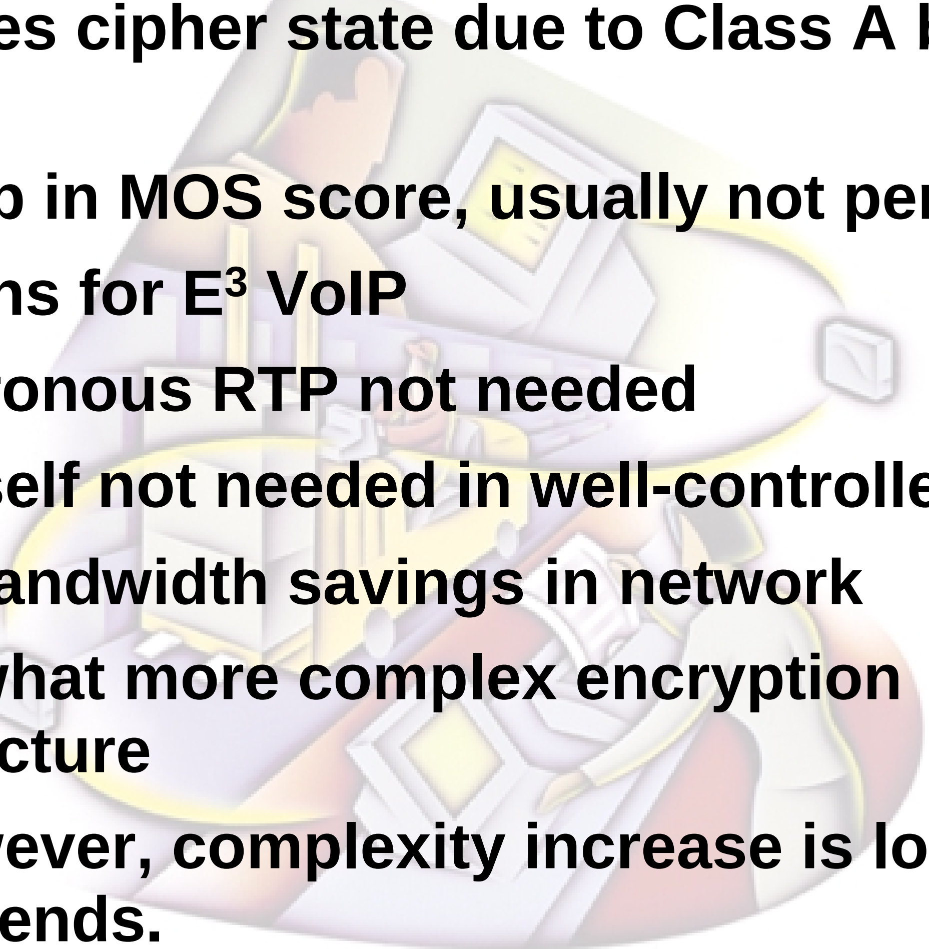
- Re-involves system during body of a call
- Problematic to maintain, especially after a handover
  - Impact: Performance degradation
- Adds complexity to network
- Needs bandwidth in network



NSE³ Architecture



NSE^3 based Properties



- *Important Advantage:* Avoids use of cryptosync, yet changes cipher state due to Class A bits' entropy
- Slight drop in MOS score, usually not perceived
- Implications for E^3 VoIP
  - Synchronous RTP not needed
  - RTP itself not needed in well-controlled VPN
    - ⇒ bandwidth savings in network
  - Somewhat more complex encryption architecture
    - However, complexity increase is localized to two ends.

How and Why NSE^3 Works

- Synergy: “Bad Frame Masking” ↔ encryption
- 3 classes of AMR coder output bits
  - Class A (Class 1a in GERAN): Protected by checksum and not used if checksum fails. Also protected by strong convolutional coding.
  - Class B (Class 1b in GERAN): Not protected by checksum, and may or may not be used if Class A checksum fails. Also protected by strong convolutional coding.
  - Class C (Class 2 in GERAN): Not protected by checksum, but used even if Class A checksum fails. Protected by weaker convolutional coding in UMTS and unprotected in GERAN. However, there is little or no perceptual degradation due to their use under checksum fail.

How and Why cont'd

- **Consider two cases: CRC Pass, and CRC Fail**

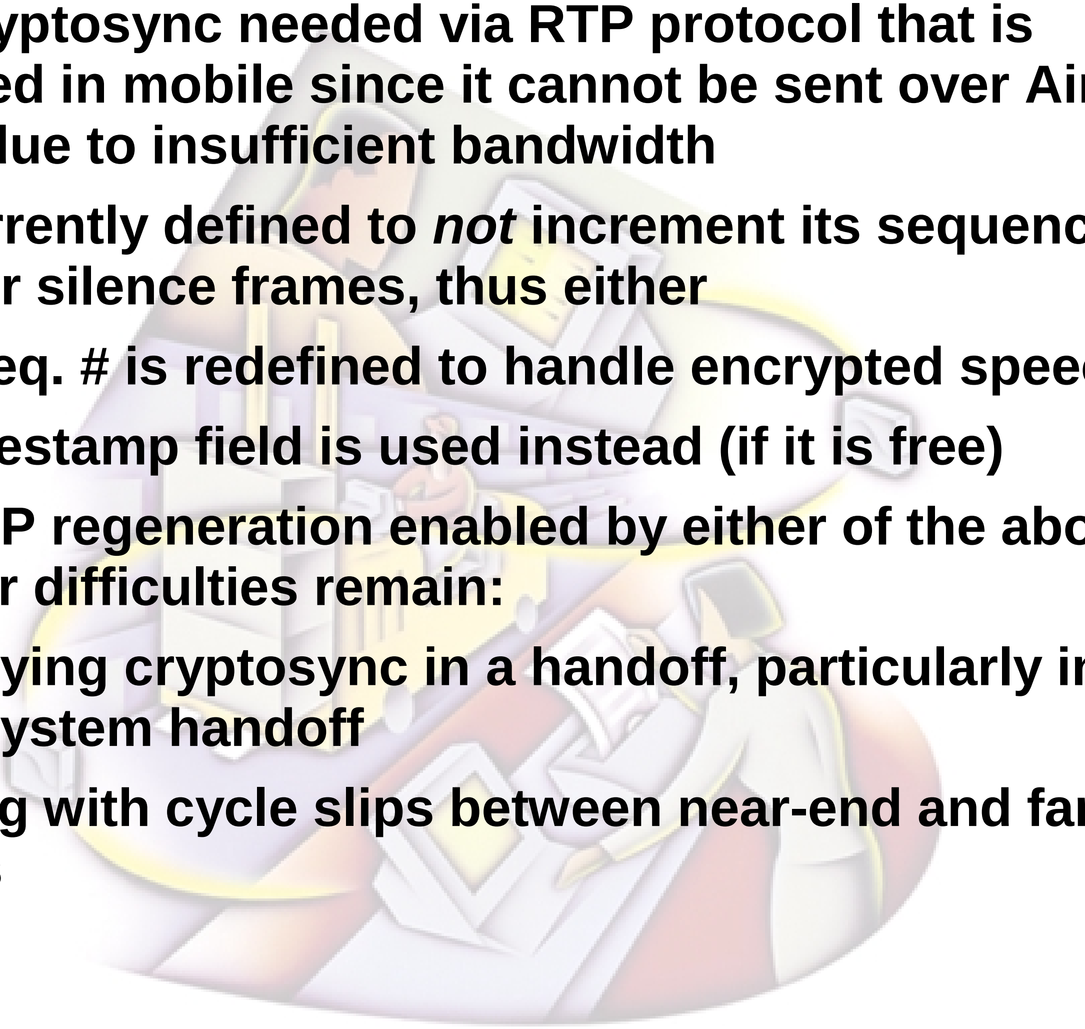
CRC Pass

- Class A bits are generally decrypted correctly.
- Since Class A ciphertext is unerrored, the stream cipher's output matches at both ends and thus no errors are added to Class B and Class C bits.
- In other words, encryption is transparent.

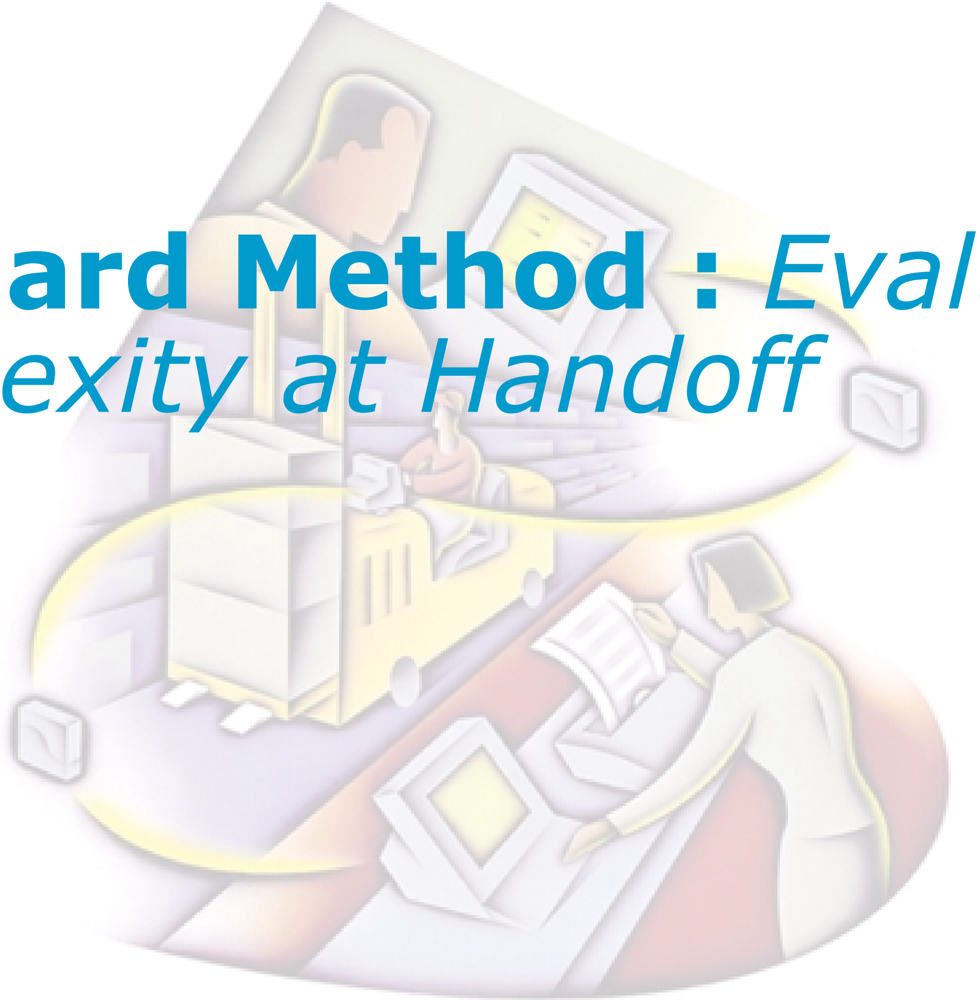
CRC Fail

- Class A and in some cases part of the Class B bits are not used from the current frame so it does not matter how their respective decryptors are performing.
- Some Class B and Class C bits are passed but totally errored by the decryption, and thus comprise random noise. However, A/B testing shows that replacing these bits by noise is not perceptible under a CRC-fail condition.
- In other words, encryption is again transparent.

Standard Method

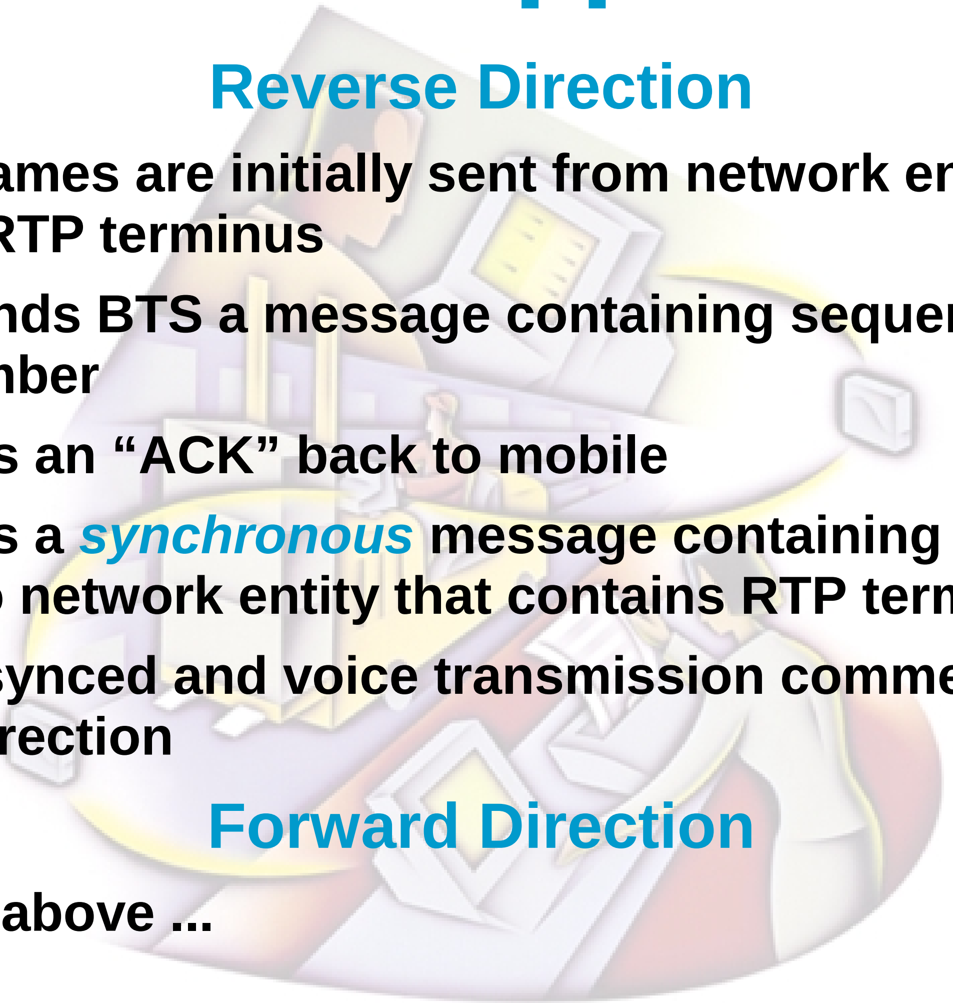


- Robust cryptosync needed via RTP protocol that is regenerated in mobile since it cannot be sent over Air Interface due to insufficient bandwidth
- RTP is currently defined to *not* increment its sequence number for silence frames, thus either
  - RTP seq. # is redefined to handle encrypted speech
  - Or timestamp field is used instead (if it is free)
- Even if RTP regeneration enabled by either of the above, two further difficulties remain:
  - Conveying cryptosync in a handoff, particularly in an inter-system handoff
  - Dealing with cycle slips between near-end and far-end clocks



Standard Method : *Evaluate Complexity at Handoff*

Resync after Handoff: A possible approach?



Reverse Direction

- Silence frames are initially sent from network entity that contains RTP terminus
- Mobile sends BTS a message containing sequence # and frame number
- BTS sends an “ACK” back to mobile
- BTS sends a *synchronous* message containing sequence number to network entity that contains RTP terminus
- RTP is resynced and voice transmission commences in reverse direction

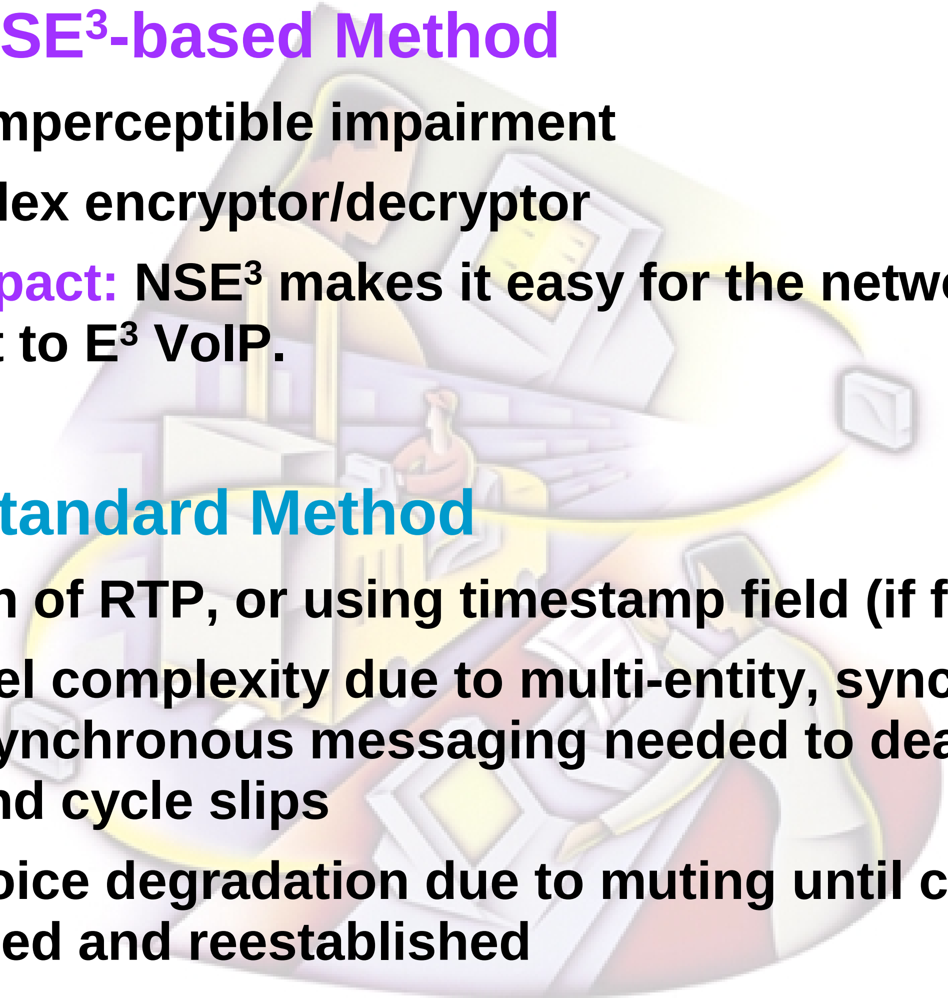
Forward Direction

- Similar to above ...

Conclusions

Impact of NSE^3-based Method

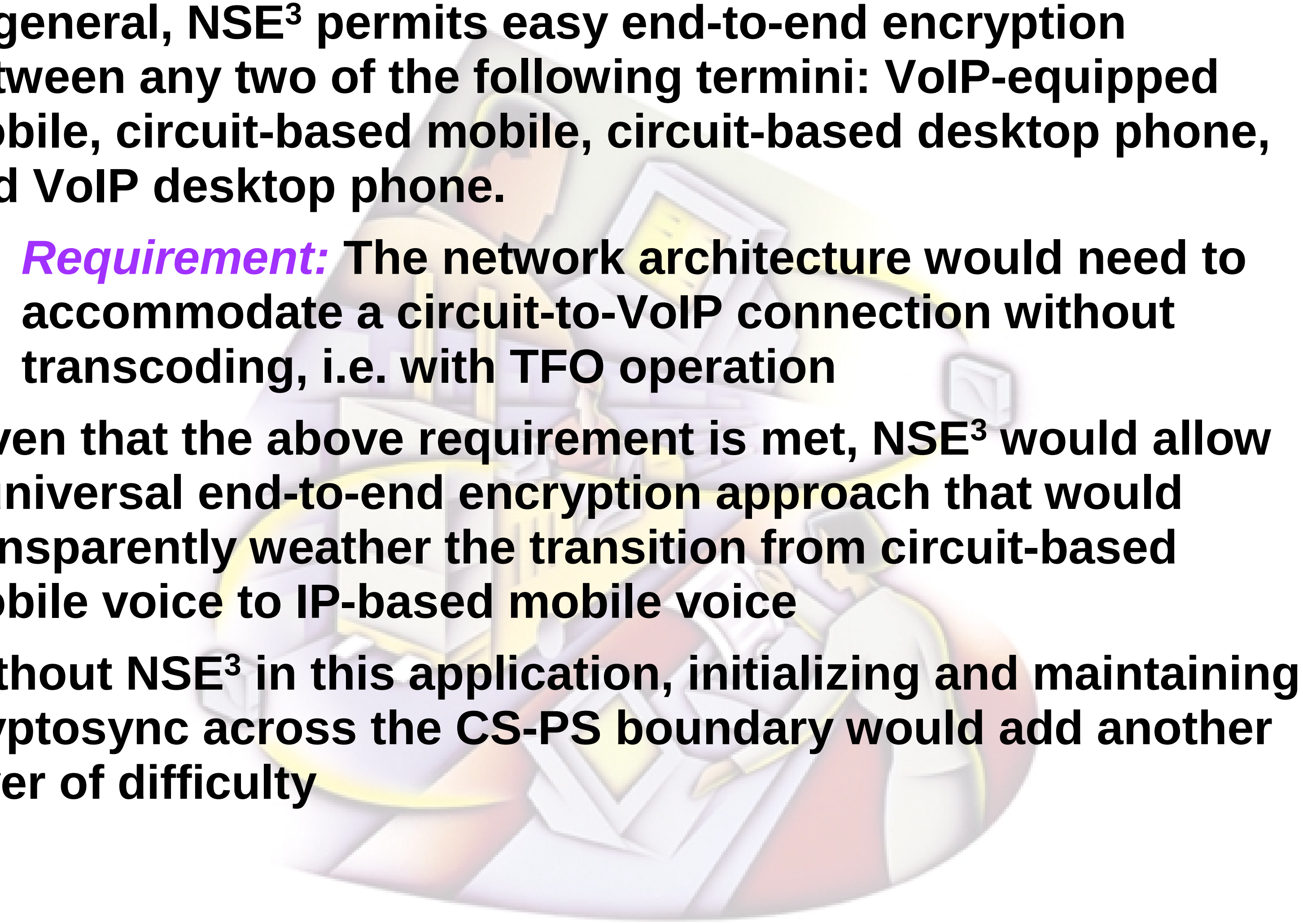
- Generally imperceptible impairment
- More complex encryptor/decryptor
- Positive impact: NSE^3 makes it easy for the network to be transparent to E^3 VoIP.



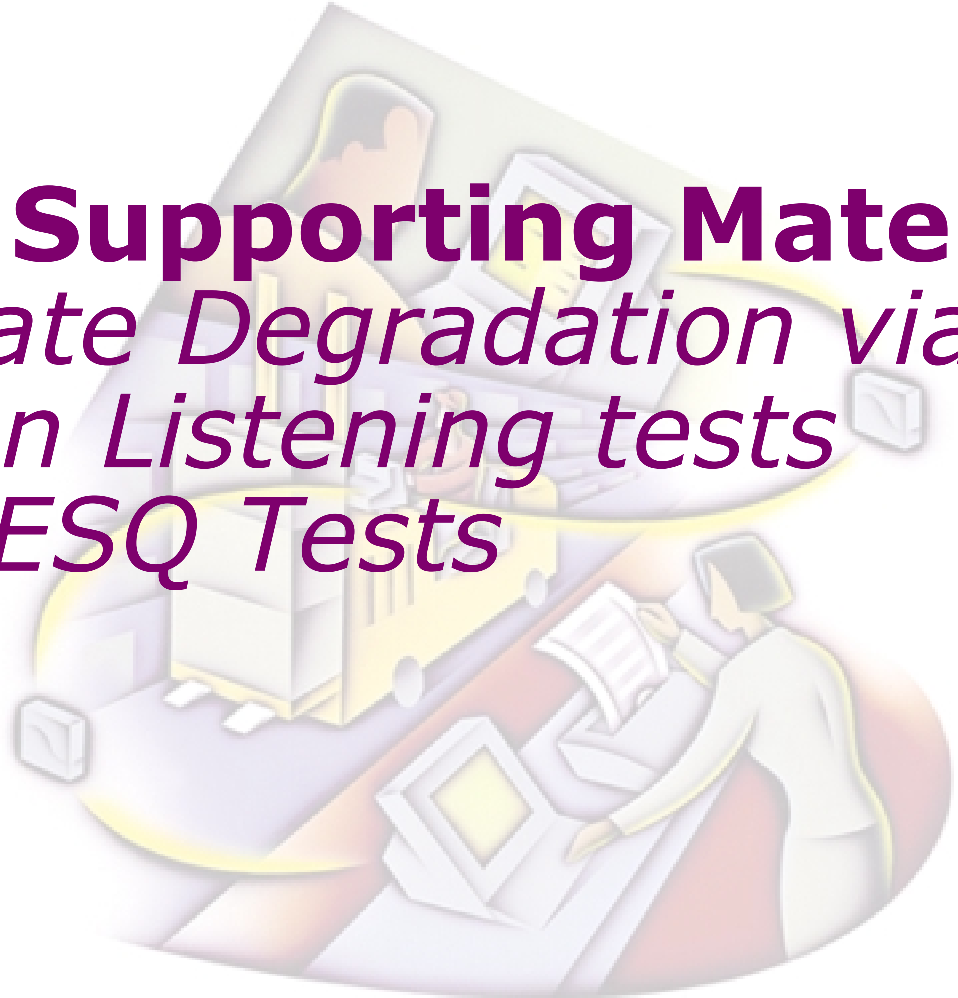
Impact of Standard Method

- Redefinition of RTP, or using timestamp field (if free)
- System-level complexity due to multi-entity, synchronous and fast asynchronous messaging needed to deal with handoffs and cycle slips
- Probable voice degradation due to muting until cryptosync is established and reestablished

Another Potential Advantage of NSE^3

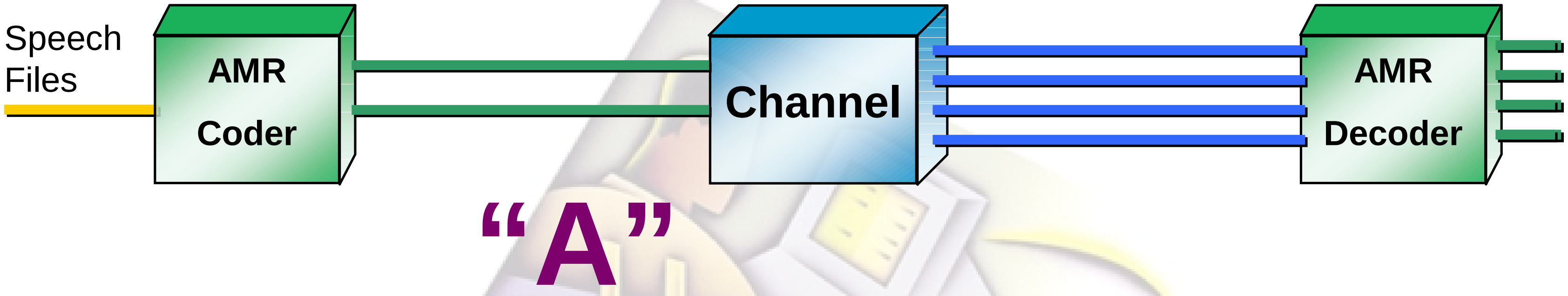


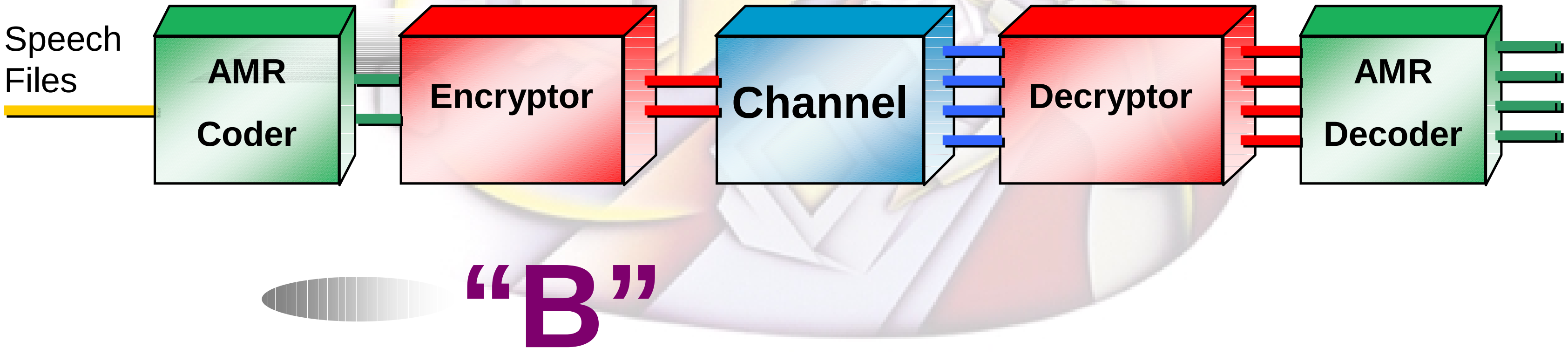
- In general, NSE^3 permits easy end-to-end encryption between any two of the following termini: VoIP-equipped mobile, circuit-based mobile, circuit-based desktop phone, and VoIP desktop phone.
  - *Requirement:* The network architecture would need to accommodate a circuit-to-VoIP connection without transcoding, i.e. with TFO operation
- Given that the above requirement is met, NSE^3 would allow a universal end-to-end encryption approach that would transparently weather the transition from circuit-based mobile voice to IP-based mobile voice
- Without NSE^3 in this application, initializing and maintaining cryptosync across the CS-PS boundary would add another layer of difficulty



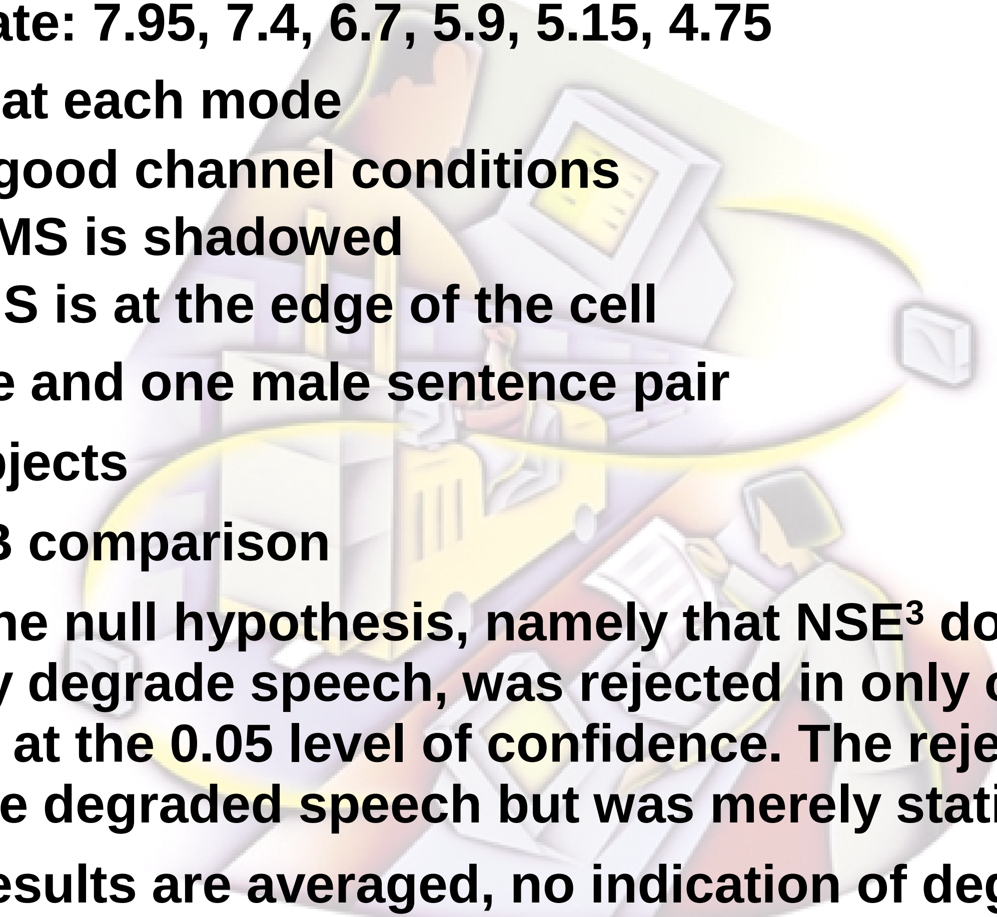
NSE^3 Supporting Material: *Evaluate Degradation via Human Listening tests and PESQ Tests*

Testing Architecture





Human Listener Tests



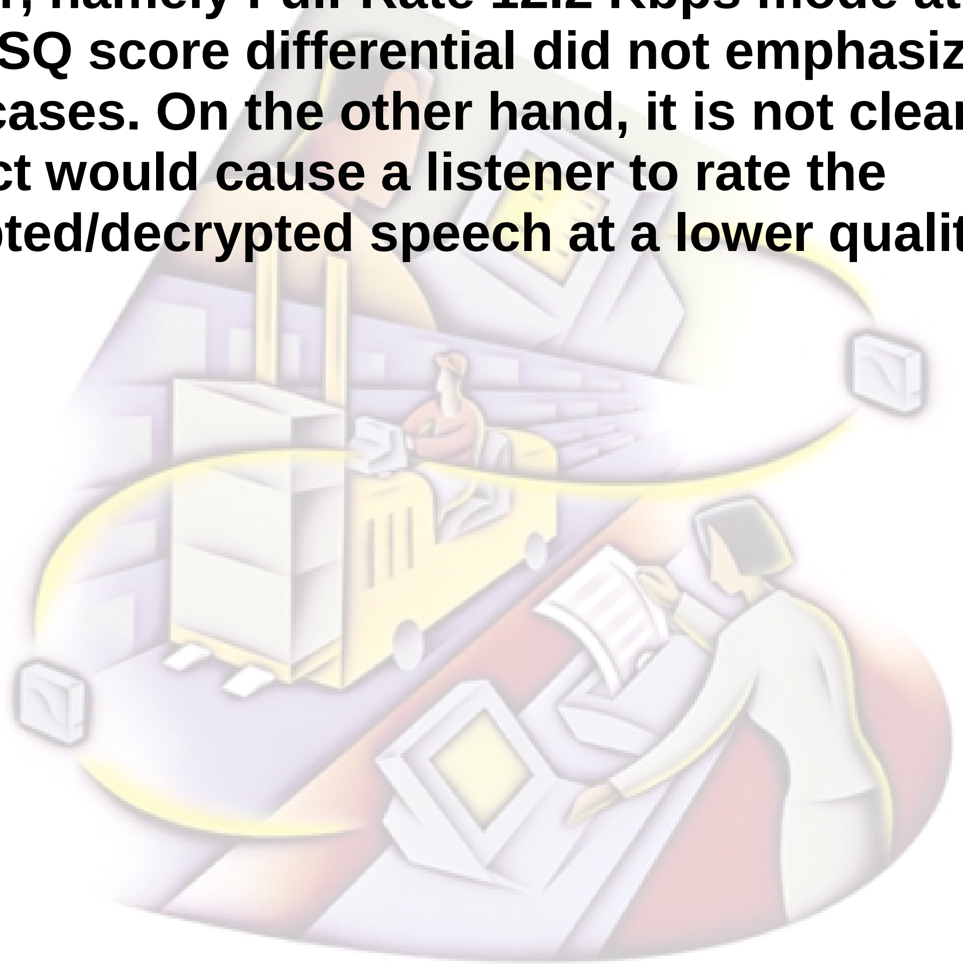
- All GERAN AMR modes
  - Full-Rate: 12.2, 10.2, 7.95, 7.4, 6.7, 5.9, 5.15, 4.75
  - Half-Rate: 7.95, 7.4, 6.7, 5.9, 5.15, 4.75
- Three C/Is at each mode
  - 16dB: good channel conditions
  - 10dB: MS is shadowed
  - 7dB: MS is at the edge of the cell
- One female and one male sentence pair
- 14 test subjects
- Forced A/B comparison
- Results: The null hypothesis, namely that NSE^3 does not perceptibly degrade speech, was rejected in only one out of 42 conditions at the 0.05 level of confidence. The rejection here did not indicate degraded speech but was merely statistical.
- When all results are averaged, no indication of degradation is present. In fact, the average is 0.7 SDs in favor of NSE^3.

PESQ Trials

- PESQ: ITU-T draft P.862, “Perceptual Evaluation of Speech Quality”
- Lucent has been evaluating the degree to which algorithmic PESQ testing models human listeners in forming MOS scores.
- Test conditions: Same AMR modes on preceding page at larger set of C/Is between 1 dB to 19 dB in 3 dB steps
- We observed several results:
  - PESQ can pick up differences smaller than human listeners’ perceptual thresholds.
  - Differences were smaller than 0.15 MOS point (threshold of perception) except in one case
  - In this case, with a difference of 0.28 for half-rate 7.4 kbps at 1 dB C/I, neither file was intelligible anyway.
  - Average difference was 0.0075 MOS point

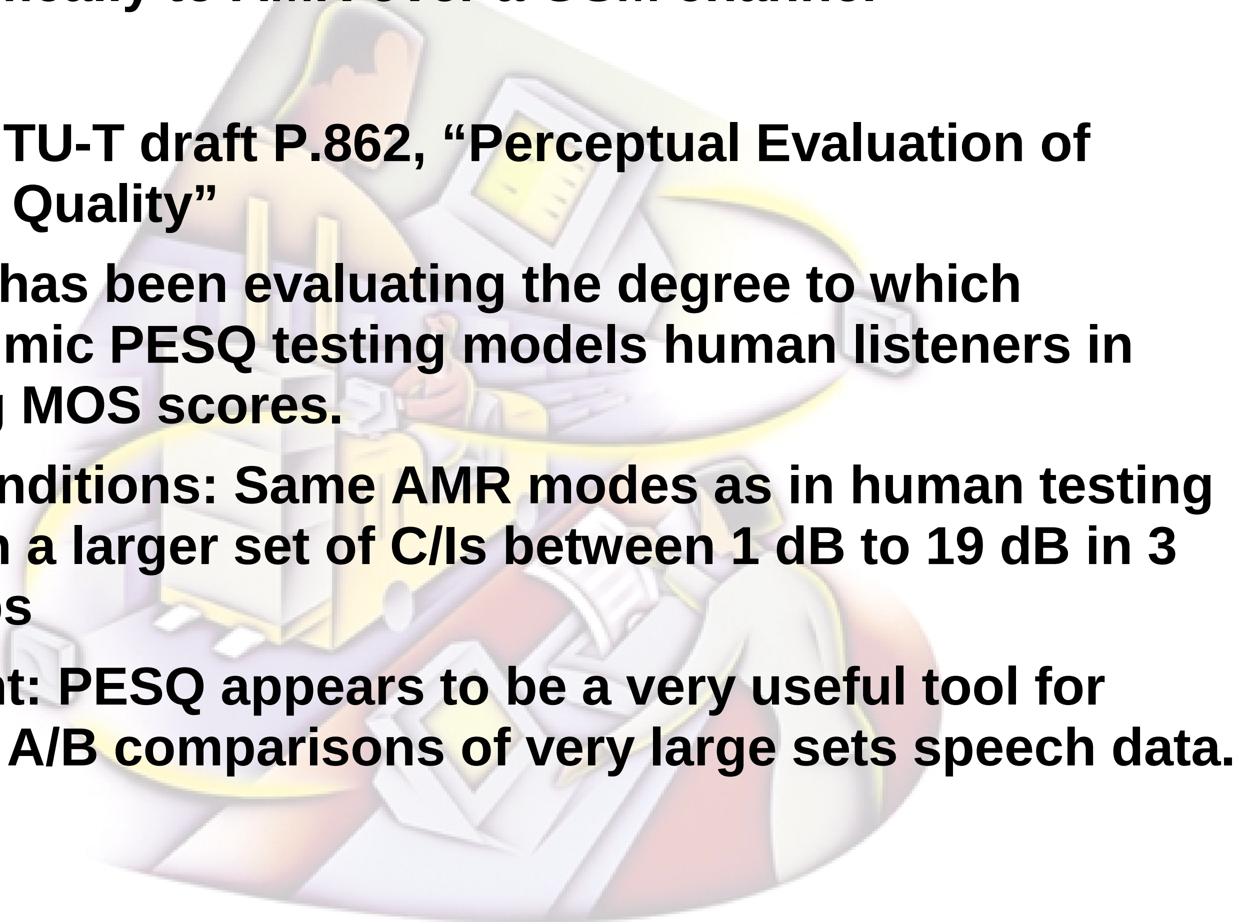
PESQ Trials cont’d

- In the one case where a difference was obvious to a listener, namely Full-Rate 12.2 Kbps mode at 4 dB C/I, the PESQ score differential did not emphasize this over other cases. On the other hand, it is not clear that such a defect would cause a listener to rate the encrypted/decrypted speech at a lower quality.



PESQ MOS Plots

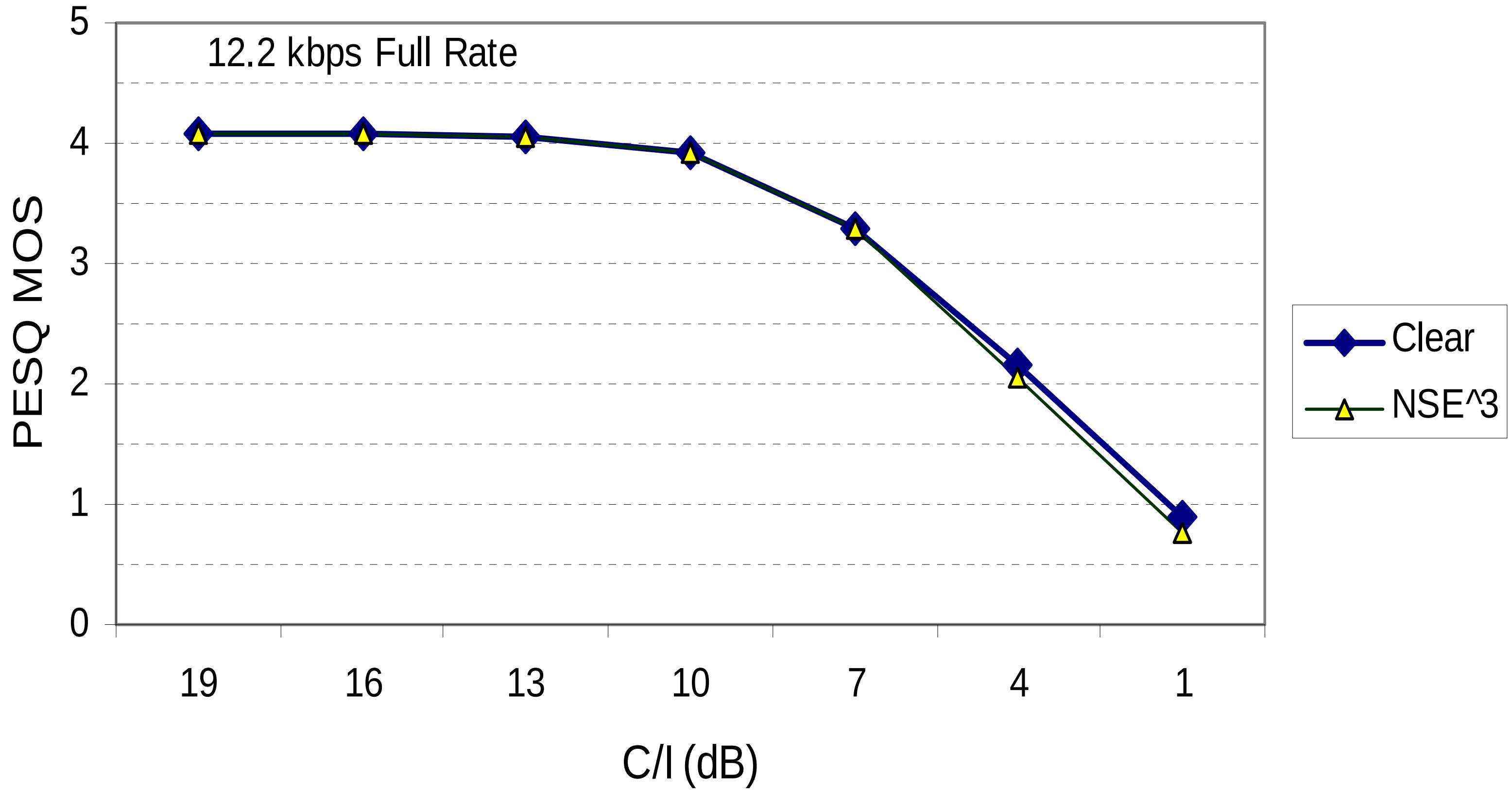
- The following plots detail the PESQ MOS results. These apply specifically to AMR over a GSM channel
- In review:
  - PESQ: ITU-T draft P.862, “Perceptual Evaluation of Speech Quality”
  - Lucent has been evaluating the degree to which algorithmic PESQ testing models human listeners in forming MOS scores.
  - Test conditions: Same AMR modes as in human testing but with a larger set of C/Is between 1 dB to 19 dB in 3 dB steps
- Assessment: PESQ appears to be a very useful tool for performing A/B comparisons of very large sets speech data.



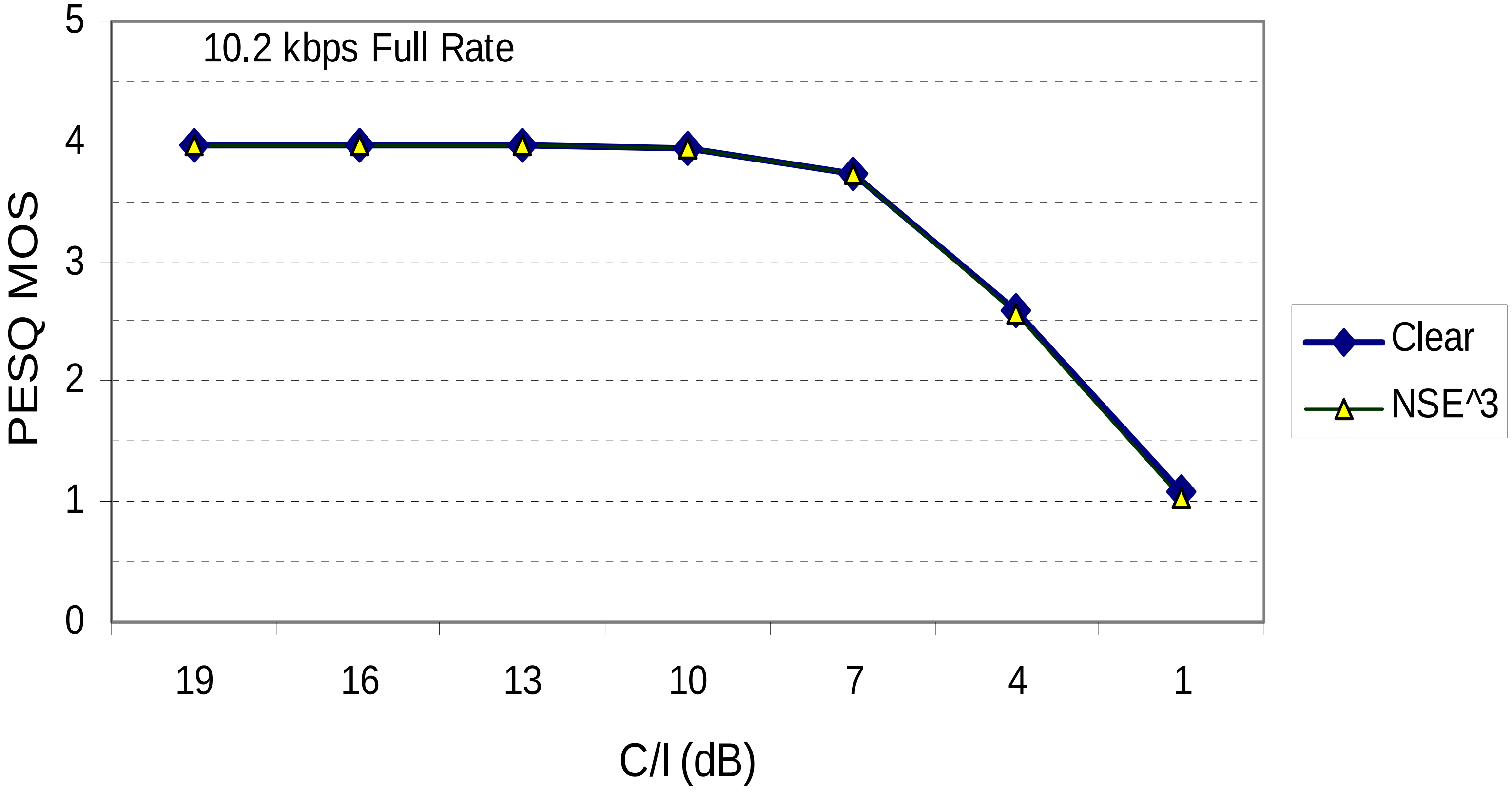
Comments on MOS Scores

- The slight MOS differences evident in some of the plots are, in all cases but one, less than the 0.15 dB perceptual threshold
- The one exception occurred at 7.4 kbps half-rate with 1 dB C/I with an MOS difference of 0.28 (See page 15.). However, neither file was intelligible anyway and the difference could not be perceived.
- The differences at lower C/Is are due to either or both of the following:
  - *False-positive CRCs:* At an equivalent frame error rate, NSE^3-induced degradation will be less in UMTS than in GERAN due to UMTS's 8-bit CRC vs. GERAN's 6-bit CRC
  - *Class 1B and Class 2 bits being replaced by random bits during bad frames:* It is possible that both UMTS and GERAN NSE^3 performance could be improved by altering the Bad Frame Masking procedure.

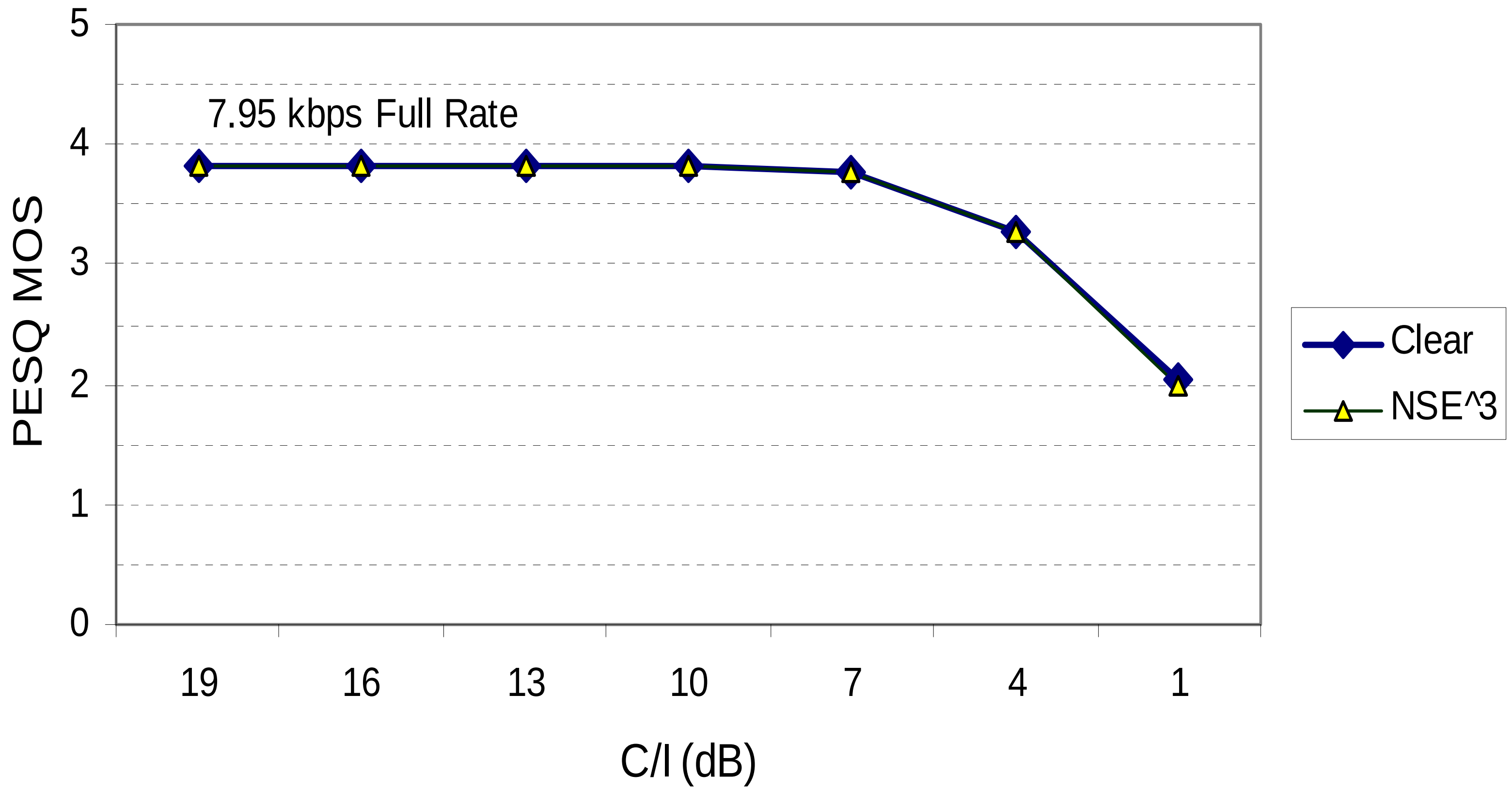
12.2 kbps Full Rate



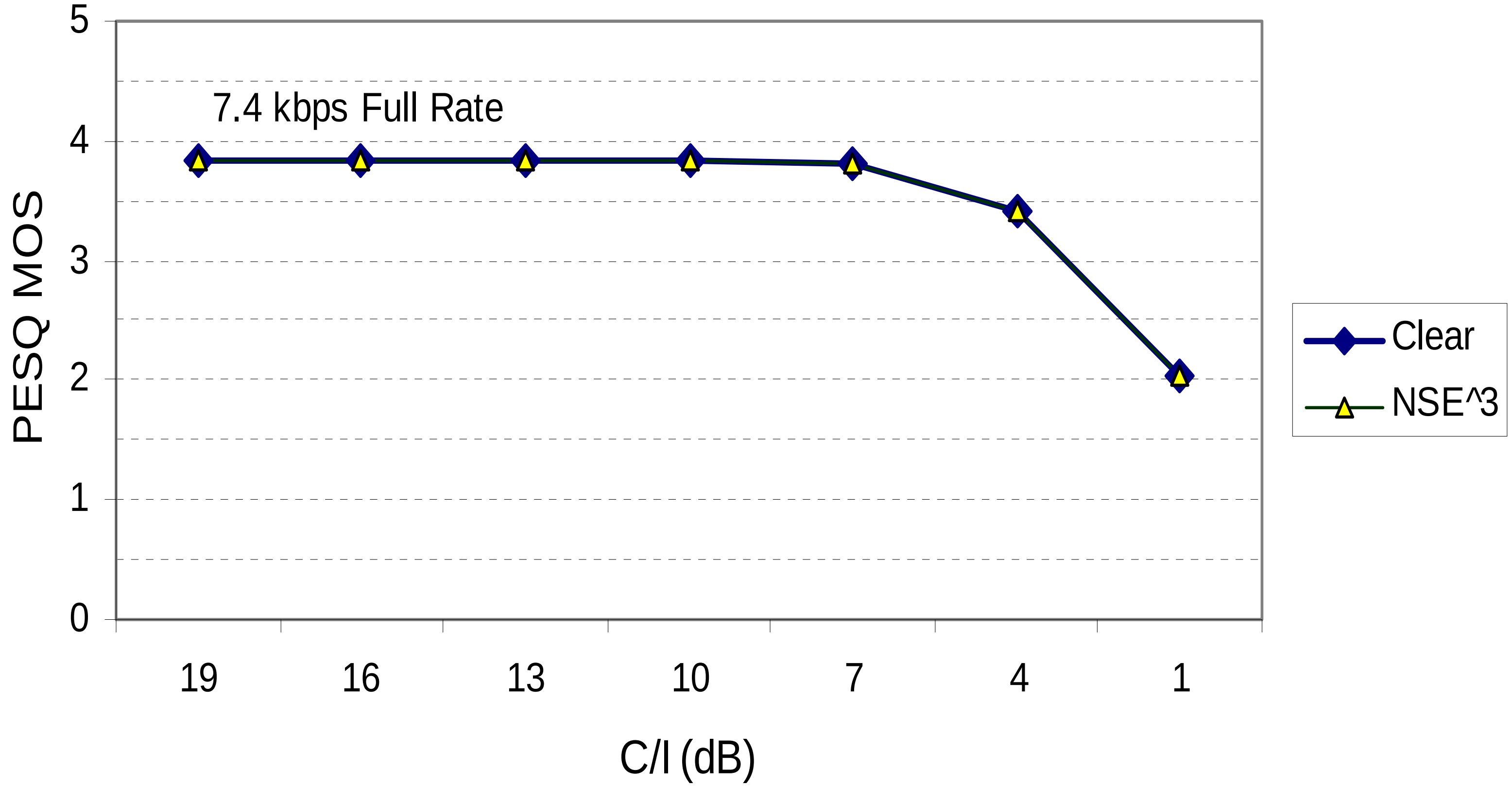
10.2 kbps Full Rate



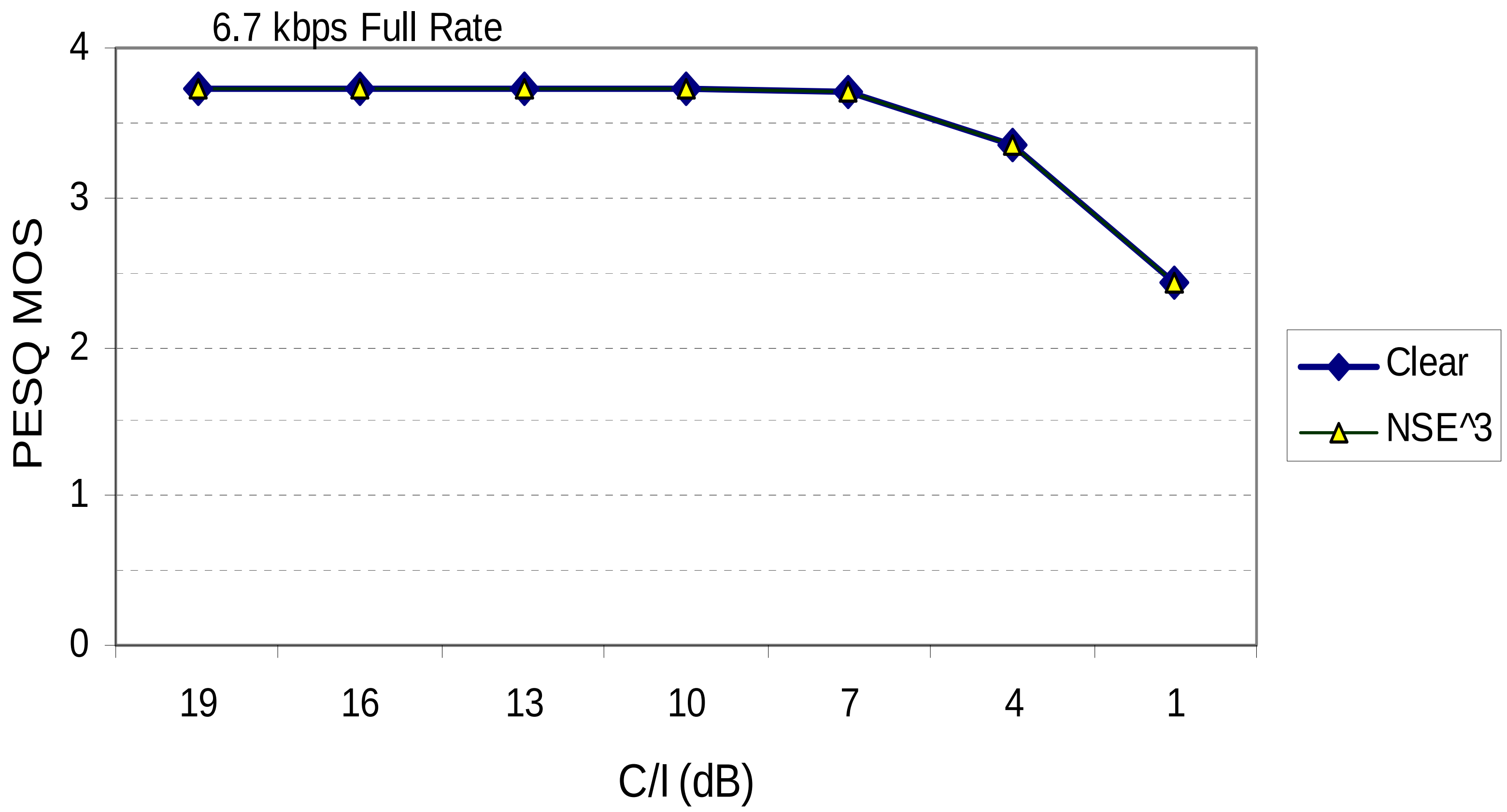
7.95 kbps Full Rate



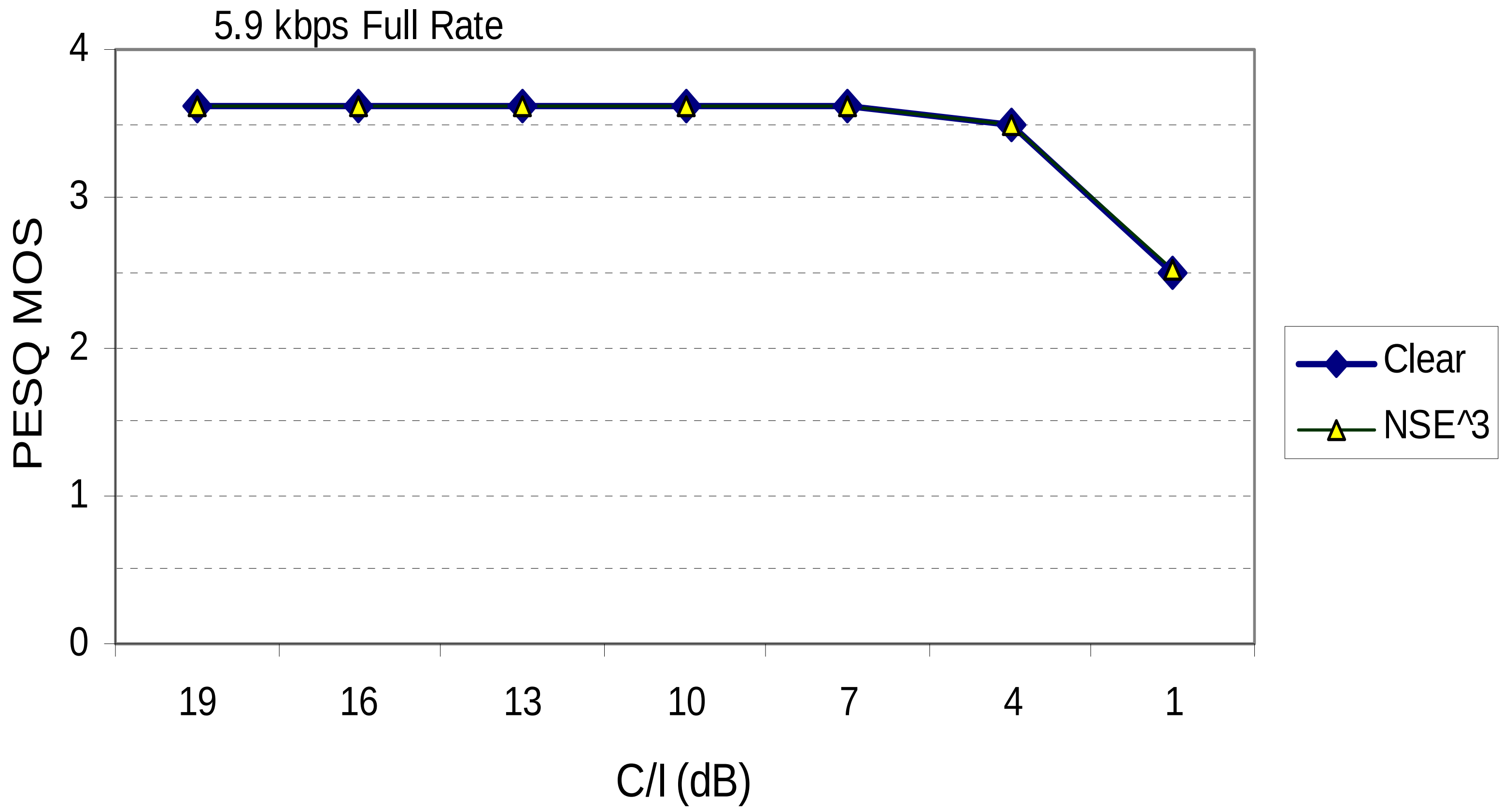
7.4 kbps Full Rate



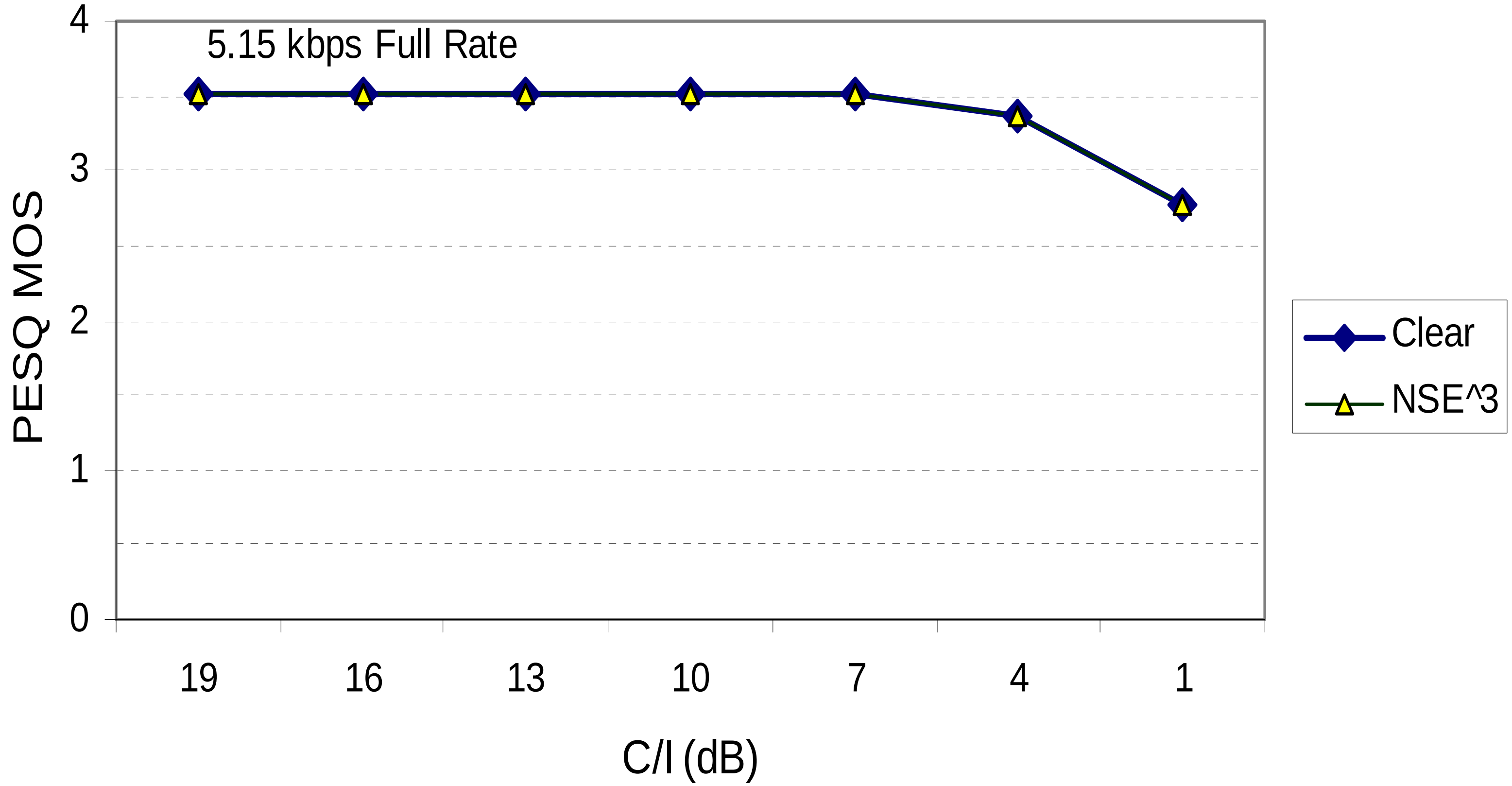
6.7 kbps Full Rate



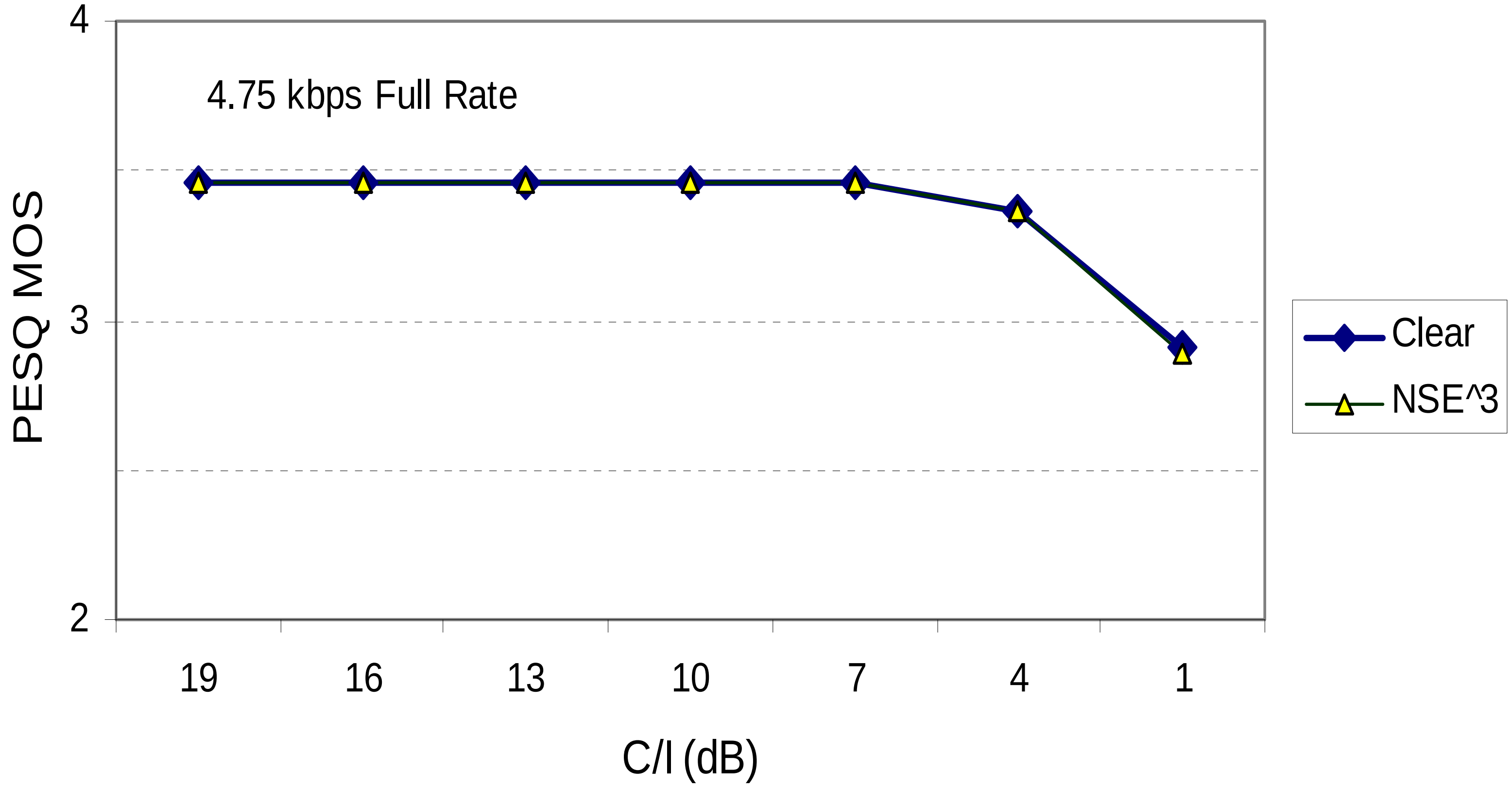
5.9 kbps Full Rate



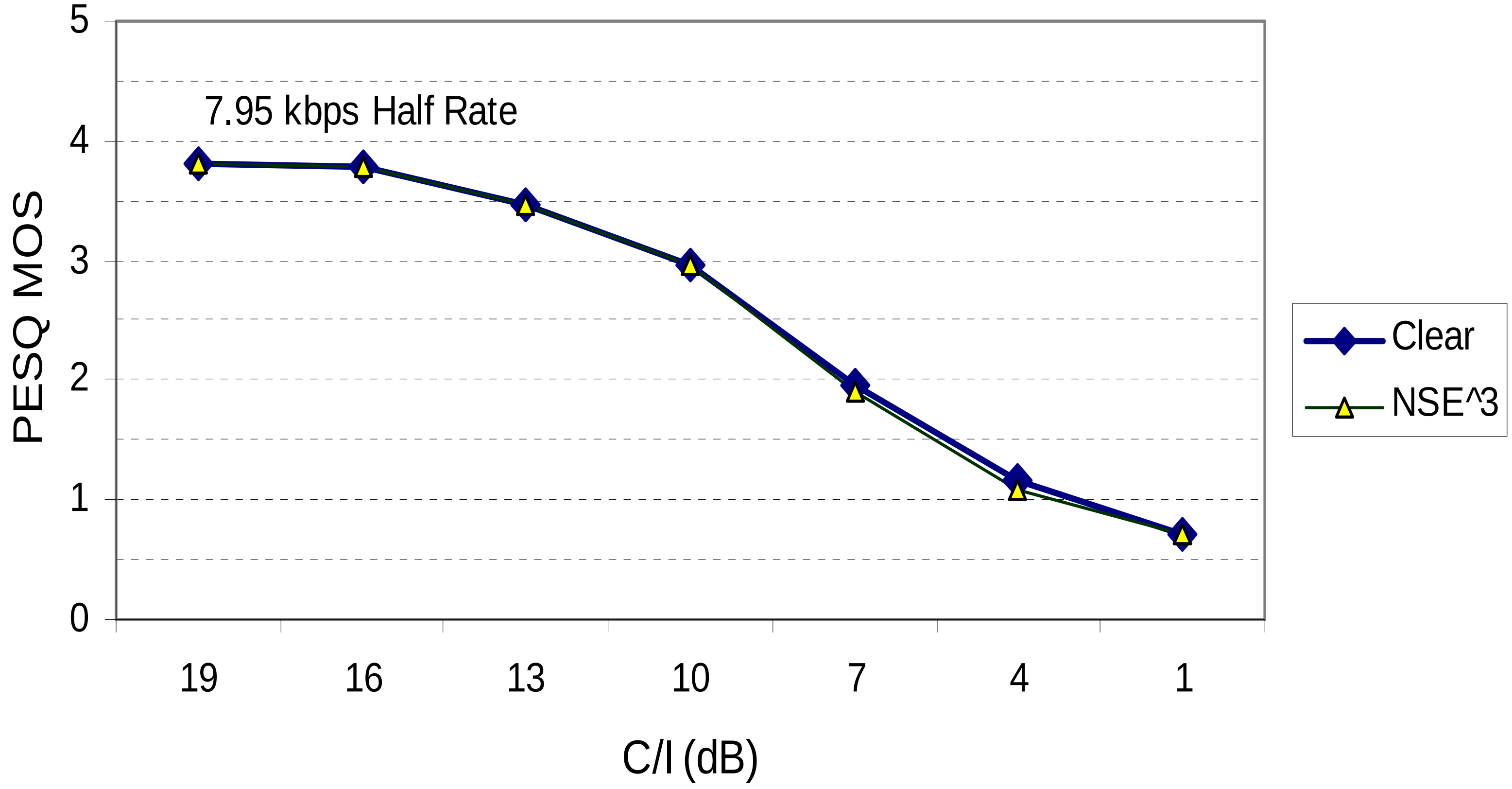
5.15 kbps Full Rate



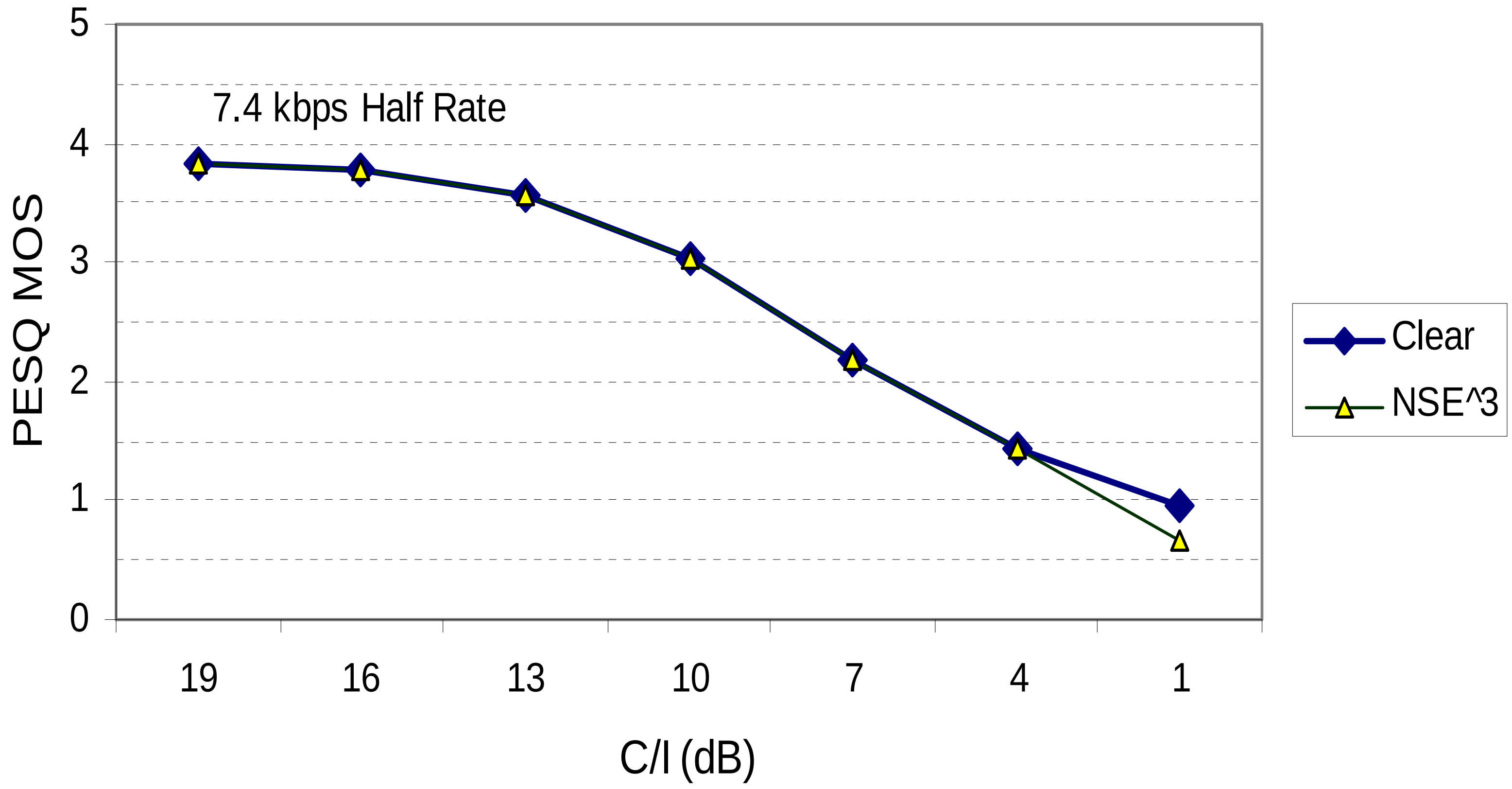
4.75 kbps Full Rate



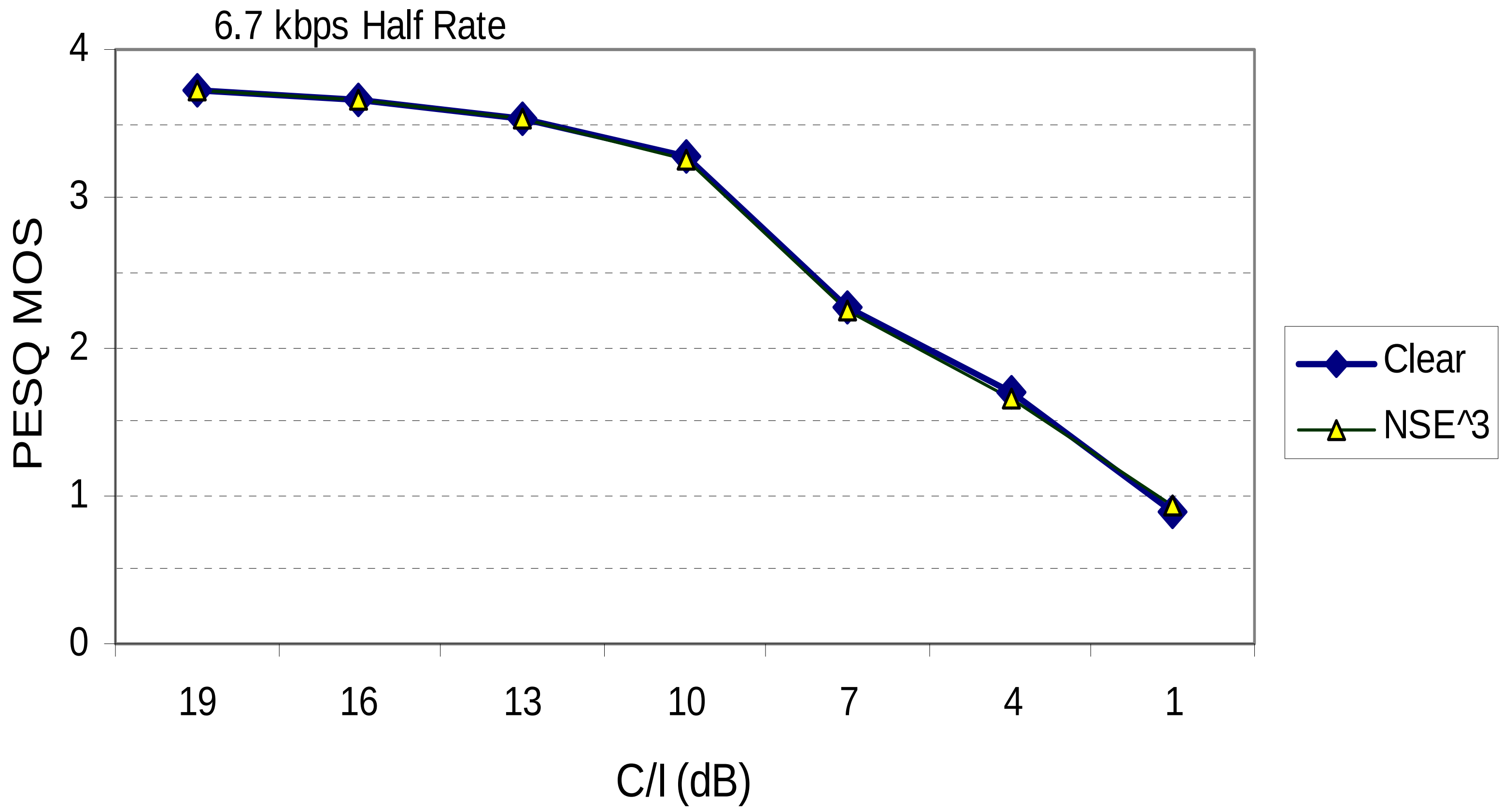
7.95 kbps Half Rate



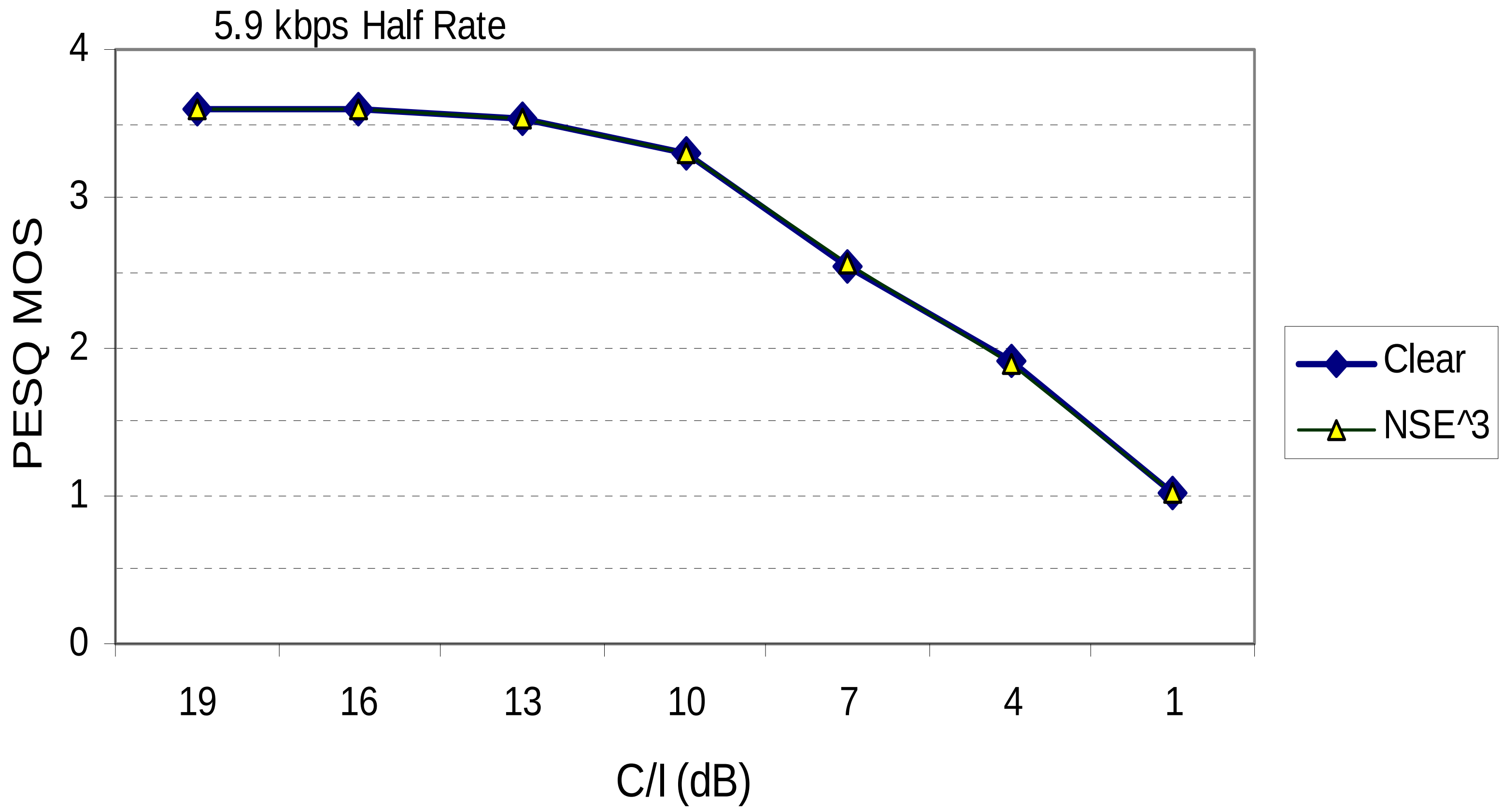
7.4 kbps Half Rate



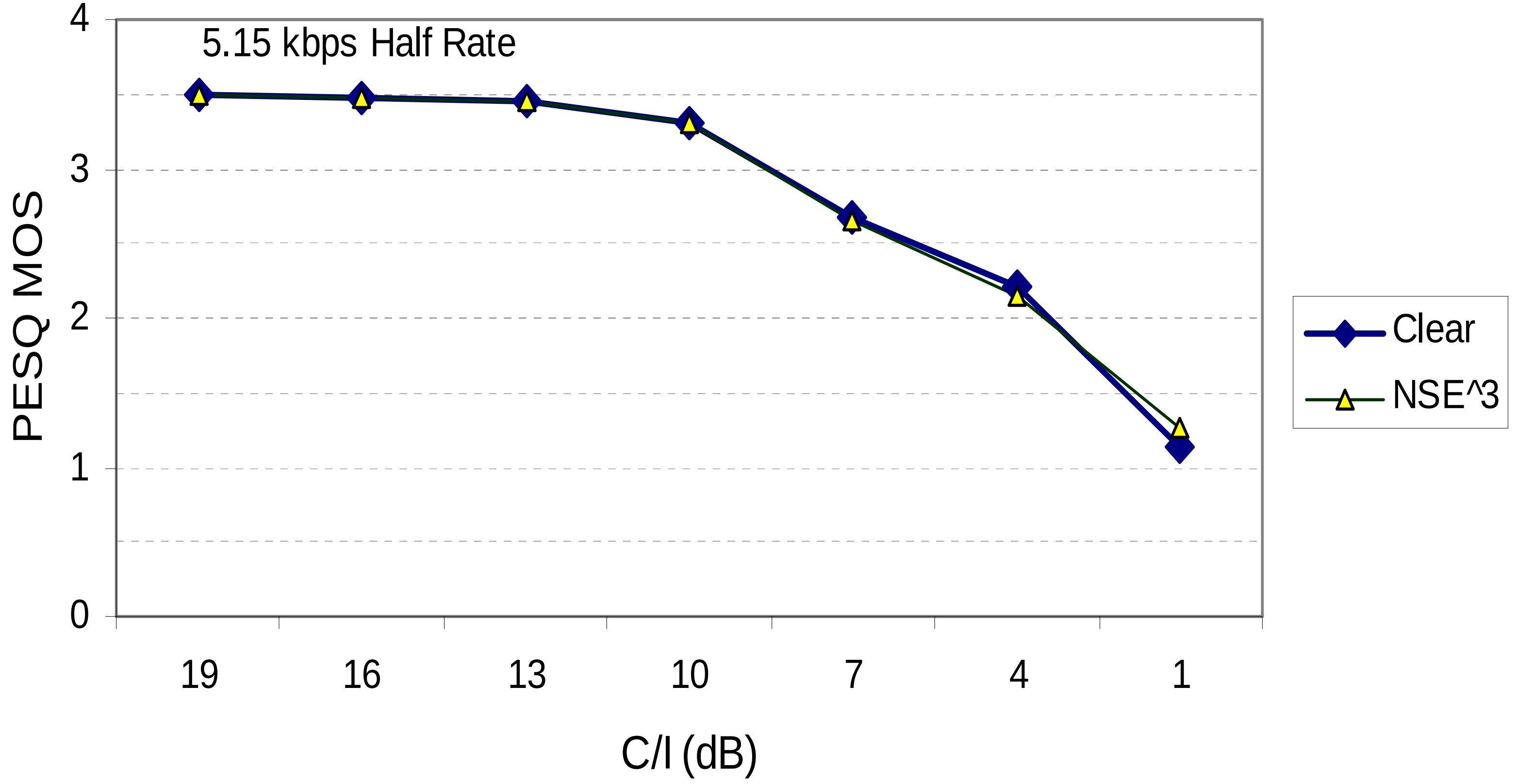
6.7 kbps Half Rate



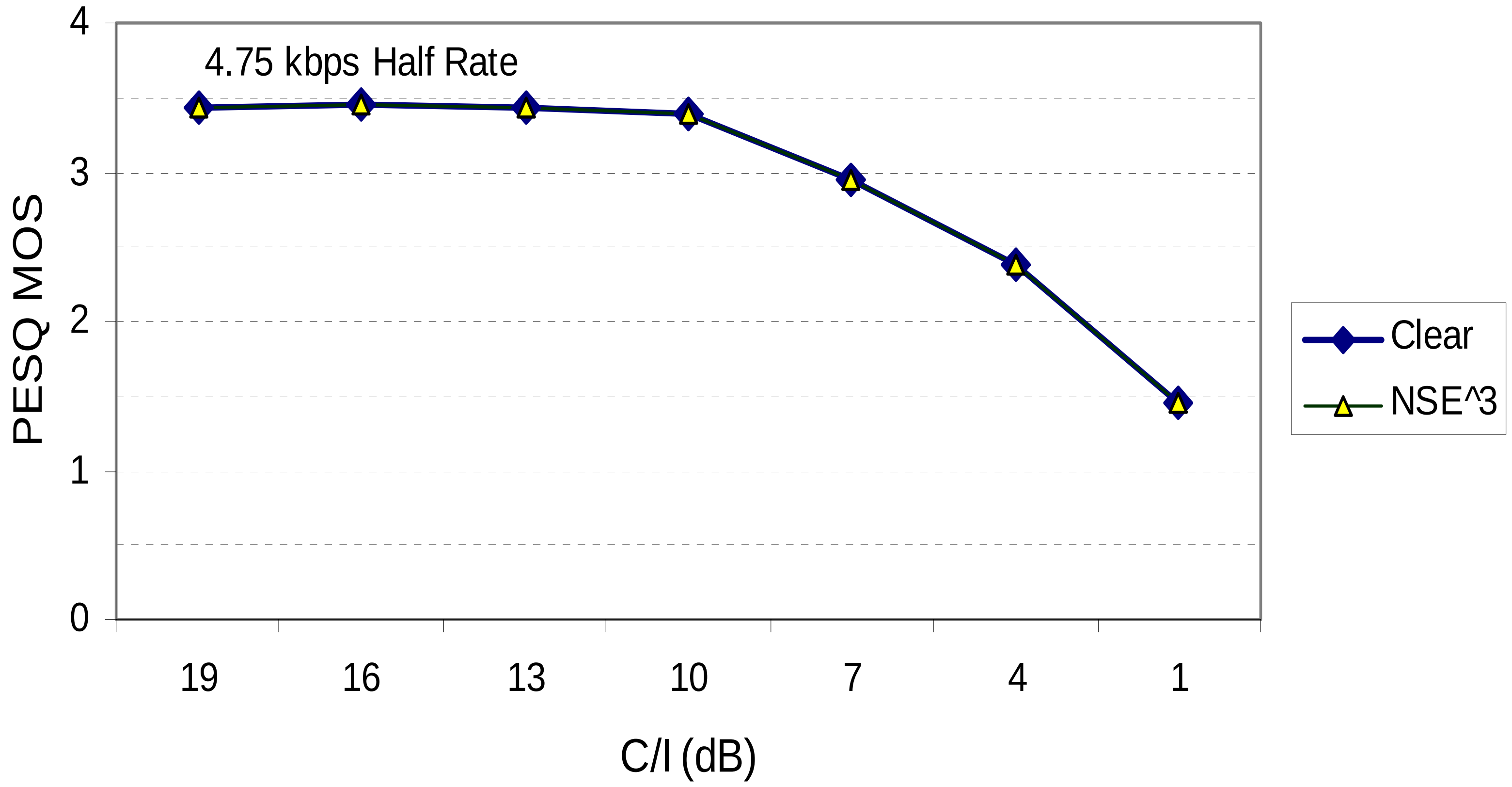
5.9 kbps Half Rate



5.15 kbps Half Rate



4.75 kbps Half Rate


4.75 kbps Half Rate
PESQ MOS
C/I (dB)
Clear
NSE^3