Proceedings of the DARPA Active Networks Conference and Exposition 2002
San Francisco, CA, May 2002, pp.427-443, IEEE Computer Society Press (unformatted version)

Resource Adaptive Netcentric Systems on Active Network: A Self-Organizing Video Stream that Auto Morphs Itself while in Transit via a Quasi-Active Network

Javed I. Khan, Seung S. Yang, Darsan Patel, Oleg Komogortsev,
Wansik Oh, Zhong Guo, Q. Gu and Patrick Mail

Media Communications and Networking Research Laboratory
Department of Computer Science
Kent State University
233 MSB, Kent, OH 44242
javed@kent.edu

Abstract

In this paper we discuss the issues that a network adaptive application faces on current network infrastructure and the role a programmable network such as the active network can play in facilitating them. We present a novel cognizant video transcoding system, which is capable of negotiating local network state based rate and let the video propagate over extreme network with highly asymmetric link and node capacities. For core operation of rate transformation it utilizes knowledge at three levels-- about the network, about the content protocol and about the content itself. As a platform the stream uses the computing power of a quasi-active network. As a result the passing video stream appears as a self-organizing stream, which automatically senses the network asymmetry and adapts itself as the packets diffuse via the active subnet. We discuss the issues of network adaptation with the exposition of this novel diffusion computing over a quasi-active network.

Key Words: Adaptive Video Streaming, Asymmetric Internet, Active Network.

1. Introduction

Adaptation is a fundamental phenomenon in natural systems. The engineering of any large and complex system intrinsically requires inbuilt ability of its components to adapt. Internet has already grown into a meganet with global reach. With the emergence of advanced applications it is now poised to evolve into a complex system of systems. With its expansion the asymmetry of the Internet is also increasing. Historically the initial Internet architecture has been conceived to cope with the heterogeneity of network standards [15]. No sooner had we thought that the problem has been caged, it now appears that a second era is evolving. Next generation of Internet will have to deal with more intrinsic heterogeneity— the asymmetry of hard network resource such as bandwidth, or switching capacity [20]. This asymmetry can evolve from the fundamental physical limitations such as the power crunch in an intergalactic network element, or from something as close and insurmountable as socio-economical disparity- the digital divide. Network technology that enables domain specific processing within network may provide novel advantages for the adaptive applications of this generation.

Commercial importance of adaptive system is growing in several areas— particularly in scalable video communication, web caching, and the content adaptation services. The advent of mobile information systems has started another wave. Due to the lack of network support, the first generation techniques are compelled to depend on resilience (by means of redundancy or indirect application level network impairment probing) to survive variation in transport characteristics, and edge processing. Less these depend on network awareness.

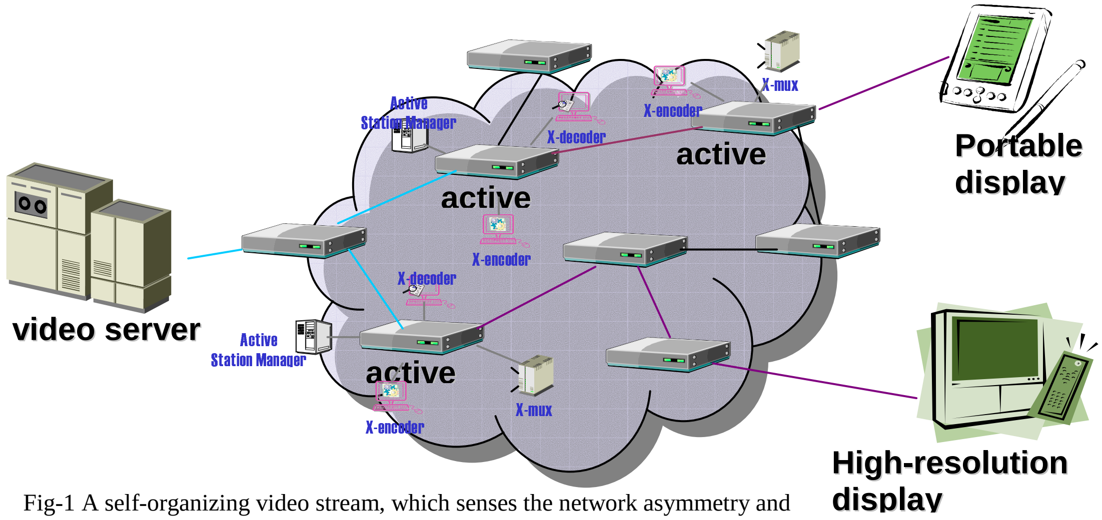

Fig-1 A self-organizing video stream, which senses the network asymmetry and accordingly adapts itself as the video packets diffuse and propagate via an quasi-active subnet. In quasi-active subnet a sparsely distributed set of routers are active in a sea of passive routers.

In this research we investigate a concept system that concentrates on creative adaptation based on the new active network technology. We present the auto-morphing MPEG-2 ISO-13818-2 [5] symbiotic video streaming system. It addresses adaptation at two levels. It adapts with respect to two critical network resources—(i) bandwidth at various links and (ii) the processing resource at the junction nodes. Though not in this form, but adaptation with respect to bandwidth has been discussed in a few other works [1,11]. However, little attention has been paid to the adaptation with respect to node capacity. The latter is uniquely important to active application paradigm by the very definition of active networking.

The key to the automorphic streaming is a mechanism for active transcoding. For link capacity adaptation, the mechanism senses local asymmetry in link capacities at various junction points of a network. Accordingly it adapts the video stream rate. In the second level, the mechanism also senses the local computation power. Based on the network computational power it first demonstrates self-organization behavior.

To adopt these adaptive behaviors it internalizes a number of techniques— some of which are novel in video transcoding as well. For example, its rate adaptation mechanism fuses re-quantization with spatio-temporal sample fusion. For extreme rate scalability it further employs fast scene analysis based perceptual encoding. To adapt with the processing power problem, instead of search, it can fast compute the motion vectors. Further when a single active node becomes insufficient, it is capable of dynamically sweeping neighboring nodes in search of increased. These techniques are novel in transcoding research. Other, recent transcoding techniques can be found in the references [7,11,12]. These vary on the degree of decoding. These are single node and quantization based.

As a computing platform this system uses a **quasi-active subnet** rather than a single active router. In quasi-active vision of active network, we point to a scenario where it is not required for all routers to be active. Rather, the proposed stream transformation is sustainable even if a few sparsely distributed routers are active in a sea of classical routers. This scenario is illustrated in figure-1. In this quasi-active network the passing video stream appears as a self-organizing stream, which automatically senses the network asymmetry and adapts itself as the packets diffuse via the active subnet. The sparsely distributed active nodes provide the computing platform required for the conversion.

While, this concept application itself is novel in its capabilities, in the process it also demonstrates architectural features required of the next generation network architecture for adaptive systems. These include polymorphic programmable service building with inheritance, dynamic network service deployment, active service management and active application/network interfacing. Overall this system will provide a glimpse into the rich and powerful next generation netcentric systems that can potentially be built on active networks.

This paper presents the architecture of this system. Before presenting the design, section 2 first explains the adaptive behaviors and the motivation behind selecting an active network infrastructure for building this system. Section 3 then describes the rate transformation techniques used by the stream and the underlying rate transcoding system. Section 4 then describes the diffusion architecture. We also identify requirements and tools, which an active network may facilitate to realize such novel adaptive behavior. Finally, section 5 shares quantitative results from the prototype implementation.

2. Pervasive Video Streaming

2.1 Location of Adaptation

There are some problems for which there may not be any effective solution in the end-to-end paradigm. Adaptation for *heterogeneous clientele* problem seems to be one of them. On a network with disparate clients with disparate link capacities, there are only three possible distribution strategies from the end-points. None of which is satisfactory. Sending only at the rate of the highest-speed stream cuts off the low speed clients. Sending at the rate of the lowest speed stream penalizes others by forcing the lowest quality to all. In a large-scale communication infrastructure the number of penalized clients under either of the strategies would always be unacceptably large. The third possible strategy (currently found in most sites) is to send multiple streams. This however burdens the network. Figure-2 illustrates the non-optimality for the *heterogeneous clientele* scenario.

It is important to understand that this non-optimality is not associated with only multicast scenario. This is a more fundamental limitation and applies to other distribution models and

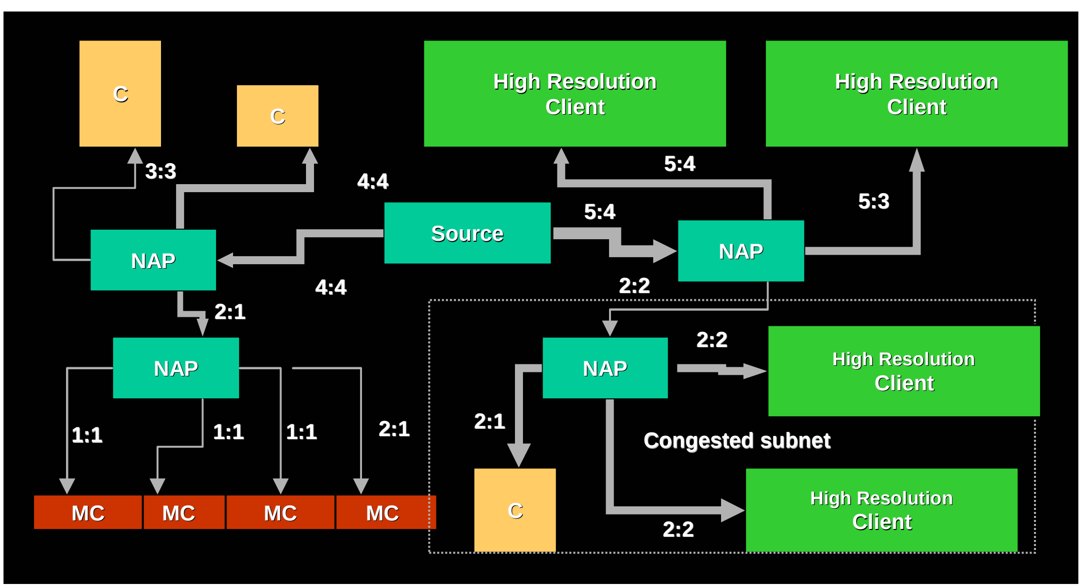

Fig-2 Video distribution in a large network with heterogeneous links and clients. Labels show the capacity and flow. Network embedded rate adaptation (NAP) points at selected junctions can minimize the net traffic and maximally satisfy the client requirements. There is apparently no end-to-end satisfactory solution to this problem. Note the distribution can be a synchronous multicast or an asynchronous store-and-forward session.

content types as well. This is intrinsically associated with the flow of information from source to multiple sinks via asymmetric infrastructure. For example if a video has to be distributed in a store-and-forward distribution model in multiple sizes, not only the link traffic, but also the cache consumption will add into the cost metric. Both the requirements will multiply with the degree of asymmetry. The same limitation applies whether the content is hypertext, interactive graphics, or a composite mix. If it is a multicast problem this limitation will increase the burden on the multicast infrastructure. If it is an asynchronous store-and-forward distribution problem, then almost in the same manner it will increase the burden on store-and-forward distribution infrastructure. The adaptation within distribution network seems to be the means to overcome this non-optimality.

2.2 Bandwidth Asymmetry and Network Embedded Computing

The bandwidth disparity between various segments of the network is increasing dramatically. The advent of non-traditional devices already demonstrates the limits of current networked applications [19, 20]. Technology is needed such that pervasive applications can be designed which will operate irrespective of the diversity and should not stop operating when encountered with network with 10-100 times lower bandwidth.

Although several attempts have been made at the networking level, but the range of rate adaptation reported was very small. It is highly unlikely that such wide range rate adaptation on the traffic generated by the demanding applications can be conveniently performed with the current network level only knowledge. When congestion arises, routers drop packets. However, current routers do not consider the application's view and treats them as unrelated items. Because of complex inter packet data dependency, dropping only 20-25% of the UDP packets can render an entire stream useless [12,2,18], though a network accounting can falsely continue to show high throughput. Effective rate adaptation requires reduction of the content, which inherently requires substantial content awareness. For video stream it requires sophisticated knowledge about content transport protocol. Unfortunately, currently content awareness is only available at the network end-points.

In adaptation it is critical to understand the value of the domain knowledge. Also, in previous sub-section we have explained the importance of the location of execution. Effective rate adaptation requires the technique to come from deep content awareness, but the resulting action to be committed in close proximity of the impairment-- both in the sense of time and space. Network embedded programmable adaptation units seems to be a potential answer to the problem. As a concept application we have therefore chosen a network embedded video rate transcoder system. A **network embedded transcoder** can solve this riddle and provide optimality even at the level of individual links, while end-to-end systems may not have any good answer to the problem.

2.3 Asymmetry in Computational Power

It seems that in near future with the rapid advancement of the VLSI technology some nodes may be able to garner enough processing power for real-time high fidelity video transcoding. However, on any given large network the fact of the matter is that there will always be inequality of processing capability just like the asymmetry in bandwidths. Consequently it is also important to develop applications, which are capable of **self-organization.** Transcoding is inherently computation intensive. It conceptually includes both decoding and encoding. A number of techniques have been investigated for accelerated transcoding by us and other researchers, such as fast DCT domain transcoding, parameter bussing [11,7,8,6]. A video transcoding is a three-way trade-off between quality-loss, rate reduction, and the computation involved. The proposed system should be built with as little possible expectation about the computational power of the underlying network elements. It should be flexible enough to choose the right operating state in this trade-off based on the network processing resources available at the junction points.

2.4 Dynamic Adaptation, Self-Organization and Seamlessness

The probability of dynamic change in the environment increases with the size of the system and the duration of the session lifetime.

Therefore, in a large and complex system the operational modules have to be built in such a way that it can respond to dynamic changes. This calls for atleast two important issues. First of all network layer services are required which can extract network local states. Also it is important that some of these be available to the adaptive elements and not only at the communication layer hosting the applications.

Adaptive elements should be built in a way that while the underlying network dynamics changes the adaptation, the reorganization is minimally disruptive to the application service. For example, if the processing power at the node changes the migration of the computation should occur with minimum disruption of the video carriage.

3. System Architecture

Recently we have built a concept transcoding system to gain understanding about network embedded programmable systems (such as active network) for adaptive complex network applications and systems. The goal of this experiment was to study an application with sufficient software and systems engineering complexity, as well as high volume data rate with temporal quality constraints, reactivity with network state, and infusion of domain specific processing. We also attempted to identify the requirements of the next generation network infrastructure.

We have chosen resource adaptation as our principal target and network embedded MPEG-2 [3,4] stream as our general application. MPEG-2 ISO-13818-2 [4] stream is the dominant open standard digital video transport protocol. It is capable of carrying video jointly compressed with multiple encoding principles with high coding efficiency. Also, the standard has been designed to carry video of wide rate class.

We have further carefully selected and implemented a set of adaptive attributes that will be representative of the various forms of adaptations and range of techniques possible in adaptive communication. We decided to include at least one technique from each major class into this experimental system to understand the principles of complex networked system's adaptation. In the top level we study (a) adaptation with respect to bandwidth asymmetry and (b) adaptation with respect to node capacity asymmetry. In each class, we have further selected sub-techniques.

For the *rate adaptation* we study the impact and mechanics of the infusion of content-awareness at two grades—(i) awareness about the **content protocol** and (ii) awareness about the **content** itself. These two techniques represent involvement of increasingly deeper domain knowledge. In content protocol aware scheme we demonstrate rate adaptation technique which involves MPEG-2 video stream structural fields. In the second level we demonstrate an even more advanced technique which uses scene analysis for greater scalability.

For the *compute power adaptation* we study the impact and mechanics of (ii) **modular self-organization** and (ii) **computation diffusion**. These two techniques represent different levels of reactivity with the network. To adapt to the available computation resource the first uses domain specific technique to cutback on the internal computation. When the capacity of a single node is too small to support the transformation the second approach tries to diffuse the computation onto the multiple nodes in the network neighborhood. The system's adaptive ability is built into two architectural layers. Below we first provide a close functional description of the transcoder, and then we discuss the systems engineering.

3.1 System Organization

We have selected a three-tier active network system where the base is constructed from converted Linux routers. In the second-tier lies the concept **active application**. The active application appears as an abstract channel with enhanced communication service provisioning capability. In the top-tier lies the **subscriber application**. The enhanced communication service provided by the active application can be subscribed and used by these top-tier applications. An example of active application is the 'Adaptive Video Transcoding (AVT) channel'. An example of subscriber application is a multimedia white-board, which needs an AVT channel for communication and distribution of its video contents.

3.2 Platform for Active Application

The scope of the research and the proposed concept system is such that many of the required supports as expected are not available in the current network infrastructure. Consequently, we have selected conceptual framework of active network as the base system, which at the moment promises ability to embed programmed capsules into network [6,9,10].

At the conceptual level, in this paper we use a **quasi-active subnet**. Instead of considering a node as the unit of activeness, in this approach we bestow activeness to the involved network as a whole rather than on its elements. An active application thus does not have to deal with individual microstructure of the network. An application views the activeness as a property of the network. The degree of activeness is determined by the mix of active to passive junction point elements, but the exact number or the exact location of the active node does not hard impact the design.

3.3 Transcoder Architecture

Logically a transcoder is a cascaded decoder and encoder. However, the re-encoding process can be made different from conventional encoding. Figure-3(a) shows the schematics of the transcoder. In full logic operation the DECODER, MV-ESTIMATOR and the XENCODER work in tandem on the stream (the other parts will be explained shortly). The overall rate control system is implemented with a dynamic piece-wise CBR (constant bit rate) rate control mechanism in XENCODER unit. Fig-3(b) shows the rate control method. Individual frame size varies based on frame type. However, it attempts to match the target rate in group-of-picture (GOP) basis. Like TM-5 [5], it is a feedback control mechanism. It accepts a target bit-rate, and dynamically changes the encoding parameters to reach the target. An MPEG-2 video stream is organized with various sample units and subunits. In the top level, the frames inside the MPEG-2 stream are organized as a *group of pictures* (GOP). The individual frames inside are then decomposed into slices. Slices generally have a sequence of macro-blocks and macro-blocks have color blocks of basic samples. The transcoder activates a number of sample processing techniques for rate reduction based on the extent of rate reduction required. First we will briefly describe the content protocol level techniques.

3.3.1 Content Protocol Aware Rate Reduction

In the first level the transcoder adjusts the quantization steps on a group of GOP basis. At the start of each GOP coding it reinitializes its target bit allocation. After each frame generation the actual bits consumed by the frame is estimated. Depending on the undershoot or overshoot the bit-balance is carry forwarded to decide the bit-budget of the next frame. Generally the carryover is discontinued after every specified number of GOPs. The quantization parameter is the lowest level parameter for controlling the bit rate. However, pure requantization based reduction [7,11] is inadequate for larger downscaling. For second stage compression we offer a sample tiling mechanism that achieves about 1/6-1/7 times rate scalability. Figure-4(a) shows the down-scalability of the system for a sample video.

3.3.2 Content Aware Rate Reduction

There is another experimental feature in this transcoder. For low bit-rate transcoding the object becomes increasingly important for perceptual quality [16]. For this case, rate control process assumes that object information (either by high level image analysis or even by direct feedback) is available about the perceptual significance of the various spatial areas of the video. Correspondingly, at low rate, bits are then taken out selectively based on the object significance. An area identified with higher significance is last to loose its bits.

In current TM-5 the activity parameter designed to account for Human Visual System (HVS) is calculated as simple variance of the pixel values. In our rate controller we have enhanced the model. The system can accept perceptual foveation information in two ways. It can explicitly accept object definition from an interface. Also recently we have integrated a motion based object processing logic module, which extracts object information from macro-block motion and accordingly modulates individual *macroblock activity* parameters of the rate controller.

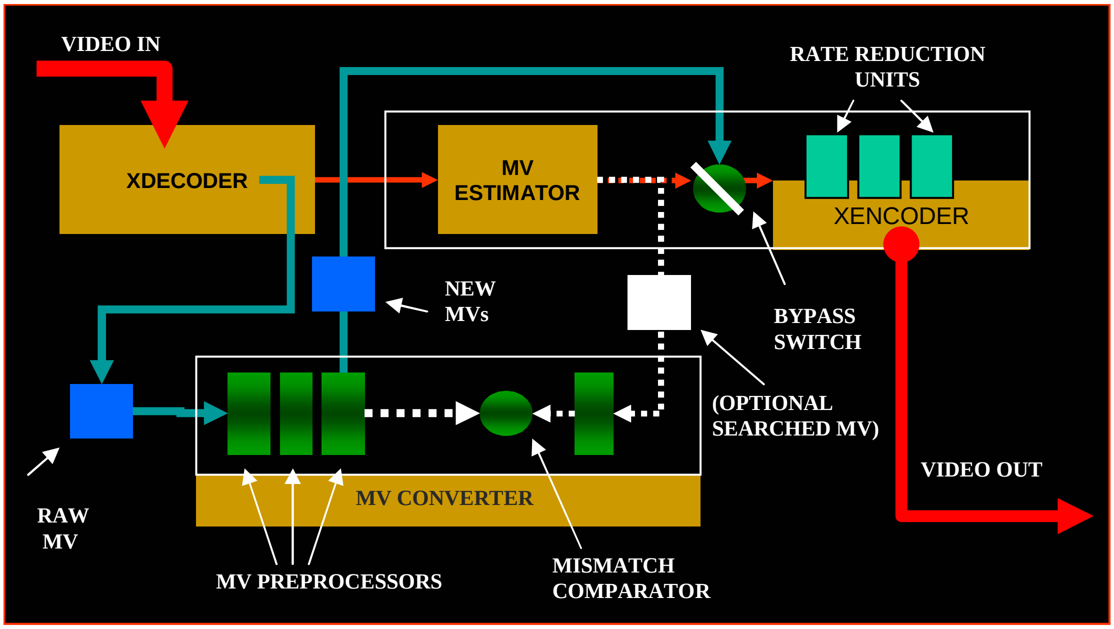

Fig-3(a) Functional diagram of the cognizant stream transcoder. For fast yet full logic rate conversion instead of searching it computes motion vectors. See Fig-8 for resulting speedup.

Fig-5(a) and (b) respectively show the difference created by this scheme. Both the frames[1] have same amount of bits but the region-wise distributions are different (area SNR shown in Fig-6(a) and (b)).

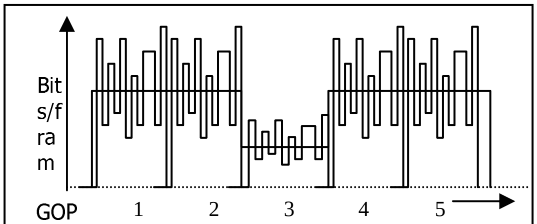

Fig-3(b) GOP wise CBR rate control with carry

3.3.3 Self-Organization for Computation Optimization

The above mechanisms provide cognizant means for enhancing the rate adaptation ability of the system. The overall computational task is nevertheless daunting. The first level of adaptation for computation power is offered by the optional motion vector [17] bypass mechanics. Figure-3 shows the functional diagram of the motion vector computation bypass capable transcoder. When switched to this mode (by enabling the BYPASS SWITCH flag), the transcoder extracts the motion vectors from the incoming stream. Instead of full search in MV ESTIMATOR unit, it them uses the extracted motion vectors to directly compute the new motion vectors. Because of the existence of the complex frame dependency and existence of the intra-frames the extracted motion vectors however cannot be used directly. We use several types of preprocessing. These are done in the MV-CONVERTER unit. It substitutes the job of the expensive motion estimation (MV ESTIMATOR) unit of full logic encoding. It accepts a series of motion processing converter plug-ins (in XENCODER). It works in parallel with the macroblock-processing pipe of the main encoder section. The bypassed motion vectors extracted from the incoming streams are fed into them. As the original macro-block data passes through tiling, temporal sub-sampling, spatial tiling, and requantization, etc. stages, similarly the motion vectors in parallel passes through a set of corresponding conversions to match the final macroblock forms. Optionally the MV-ESTIMATOR can be turned on and the mismatch amount can be monitored in MISMATCH COMPARATOR. Figure-8 shows the speedup due to the motion vector computation bypass

[1] More details appears in the center. Compare the face, the white jaw line of the whale, and the tone of the water.

mode of operation based on the search space specification it reduces 25-50% computation.

3.3.4 Diffusion Transcoding

Self-organization works as a means for adaptive tradeoff between coding efficiency (compression ratio vs. picture distortion) and speed of transcoding. However, it is quite possible that even after the reduction a single node may not be adequate to support a particular computation. A second level of complex adaptation is performed by another novel aspect of this system--computation migration. If the processing power available/allocated to the stream at the current network point becomes inadequate then part of the transcoding operation can be further migrated into multiple network nodes.

This is achieved by modular design of the transcoding units. The internal transcoder system has been formulated with a rather unique view so that the entire transformation (and the associated transcoding operation) on a video stream during its passages can diffusely take place on an active subnet rather than on a single network point. The key to the design is the modular decomposition of the transcoding operation—where instead of a single monolithic implementation we took an experimental approach of building them as dynamic hyper-linkable capsules with easily separable data flow optimized concurrent modules. Within the main units we similarly allowed for dynamic hyper-linkable plug-ins.

In our current implementation the main units are (a) GOP-Encoder (GOP-ENC), (b) decode-demultiplexer (DE-DEMUX) and (c) GOP multiplexer (GOP-MUX). The GOP-ENC is the XCODER of Fig-3(a). One GOP-DEMUX and one GOP-MUX are activated at the designated sub-net entry and exit points. However, multiple instances of the GOP-ENC units are then kept in a dormant state in designated active nodes for seamlessness. Active modules are then connected by a feedback system. The GOP-MUX module senses the resultant frame rate. If the current aggregation is less than a given target frame rate, it then sends signals to DE-DEMUX unit, which then activates dormant GOP-ENC units in the active subnet to join in.

4. Supporting Components

In this paper we have focused on the adaptive elements and techniques of the channel. For its operation as a channel it also requires a number of regular auxiliary system management services. A number of issues related to programmability, capsule deployment, remote loading and execution have been discussed in the recent active net literature [22,23,24]. Detail of a transcoder stream deployment mechanism is in [14].

Here we discuss some of the issues related to complex applications deployment. We briefly describe the methods we used[2]. This will provide important glimpse into the type of algorithms and techniques which have to be developed and/or facilitated at systems level to support network adaptation.

4.1.1 Topology and QON Search

This includes (i) a mechanism for network topology discovery. (ii) A tool for initial and dynamic tracking of selectable *quality of network* (QON) parameters. The QON requires access to link and junction element statistics. These will also help in initial and subsequent redeployment of the channel components. For the case of transcoder, we needed statistics about peak and available processing capacity at the active nodes. Similarly, we needed the peak and current traffic load at the virtual links (can have multiple hops of passive routers) between the active routers. For rate tracking and jitter control, we needed delay statistics.

The frequency of the runtime probing was GOP basis (once or twice in every second). Apparently most adaptive applications linked with perceptual data communication may be satisfied with similar probing frequency resulting in small probing traffic. Topology and QON probing is expected to be very common task for adaptive applications. Thus, some form of "QON" propagation in a manner "routes" are propagated today, might be provisioned at system level.

4.1.2 Group Communicationware

The channel components also required short but regular signaling, such as frame rate measurement. We have developed a group communication ware called "Harness" [21] to support customizable but patterned

[2] Optimized and scalable versions of these tools will be needed as a system level tool for adaptive systems framework.

communication. This is also an active application by design. It enables a group of components to exchange information along various network topologies in about 2-10 second interval on the current Internet. It isolates the communication from the content. The pattern of communication can be separately configured than the content. The content can be varied by programmable -message aggregators and synthesizers at junction points. The aggregation enables it to scale. Details about the "Harness" can be found in reference [21].

While the Harness is a specific example, adaptive network systems engineering will require powerful group communication tools, and this too has to be facilitated in active network systems.

4.1.3 Transcoder Deployment

The placement problem is divided into two stages. The first is the determination of the *constriction points*—of the approximate locality where the rate transcoding has to be performed. The second stage is the casting of the transcoder elements in the neighborhood of the constriction point. The placement logic required the approximate network topology and the QON descriptions of the involved links and nodes. The approximate topology search algorithm searched for the distribution tree between the designated source and sinks. We have given an "active harness" based algorithm for determining the constriction points in [21].

4.1.4 Component Mapping

Once, the constriction points are determined the component deployment process requires finer network map nearby. The algorithm runs a k-best path search algorithm between the source and constriction-point. We then further run a k-radius neighborhood search to include few additional nodes into the picture. It then accepts the *component connection map* [4]. This is a description of the constraints on how the components have to be topologically connected. The component connection map serves as a partial ordered list. According to the topological sorting of the component connection map, the components are assigned backward from the point of constriction. In any active network system, such dynamic casting of capsules based on capsule's requirement and the available network's resource will increasingly become a routine task worth making it another system level service.

4.1.5 Coordination

One of the complex tasks here is the coordination. To keep it manageable in this experimental implementation we maintain a single *Meta-Controller* (MC), which is responsible for user interaction, system initiation and dismantling and setting up initial boundary state configurations. All network state information is collected locally, but is relayed to the MC. MC converts the incoming states into global operational state and generates appropriate control signals. The coordination between various modules is performed by two parameter sets. The local parameters are visible only to individual instances of the units, and the MC. The global parameters are visible to all units but only the MC can modify it.

The centralized architecture seems to be sufficient for the co-ordination of the simple transcoder. It is also desirable because a centralized view is important for facilitating its use. However, there are potential pitfalls. The run time operation generally will involve much smaller number of components. There may not be any bottleneck in this part of the task. However, for dynamic adaptation a much larger pool of nodes and links have to be tracked. Scalability will be important in this part of the task.

5. Experiment Results

The design goal of this system was concept demonstration. The implemented prototype is called *Self-Organizing Network Embedded Transcoder* (SONET). For experiment we have placed it on a simple asymmetric network with a quasi-active subnet segment. We placed an MPEG-2 server in a high-speed segment and a simple MPEG-2 client into the wireless network. The network in-between contained a small quasi-active subnet. Here we share some experiment results of this system. This concept proto-type performance is given for its user space execution. In this overall system there are numerous opportunities for quantitative optimization particularly in the speed aspects. However, these figures still should give valuable insight about the computational capabilities that might be sustainable on an active channel system.

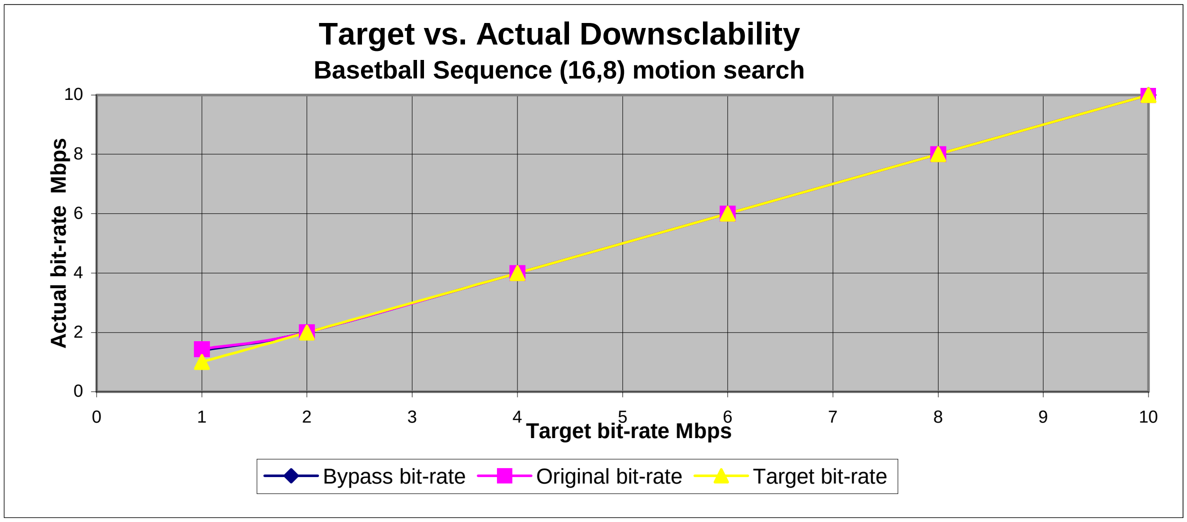

Fig-4(a) The generated rates closely following the target rate specification. The transcoder can offer rate scalability upto 1/6th-1/7th of the original bit stream.

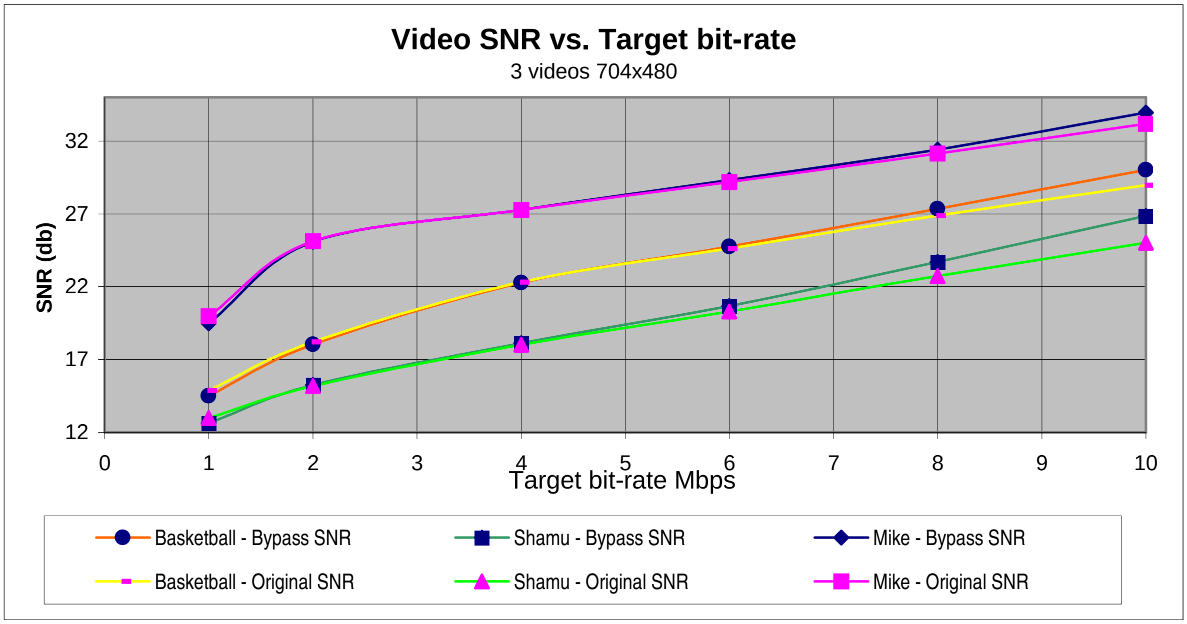

Fig-4(b) The quality distortion at various target rates for three video sequences. The motion vector computation bypassed mode reaches performance very close to that of full motion search.

Fig-5(a) rate transcoded output with uniform perceptual encoding using conventional TM-5 quantization.

Fig-5(b) rate transcoded output with object analysis based perceptual encoding. Note the detail in human face, whale's teeth line area. Perceptual encoding requires content awareness, but can produce superior perceptual quality images given same bit rate.

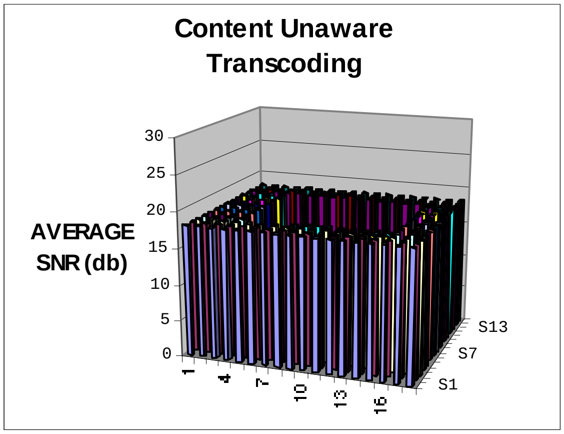

Fig-6(a) The SNR plot without perceptual encoding. Corresponding macroblock wise SNR distribution trace after simple quantized encoding. Note, the content unaware scheme can miss serious quality at the perceptually most important area due to the high activity there.

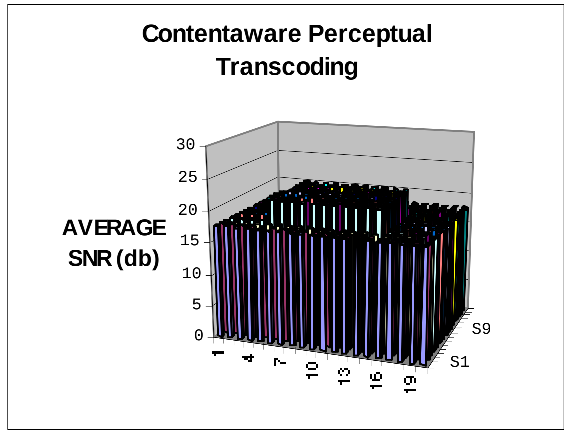

Fig-6(b) The SNR plot with perceptual encoding. The perceptual encoding reversed the tendency and increased the quality at the center where perceptual significance is higher. It took away the extra bits from background.

5.1 Scalability and Domain Knowledge

Figure-4(a) first shows the range of video (down) scalability using quantization-based rate adaptation from the content protocol aware suit. It plots the target-bit rate and the achieved bit rate in both full motion computation (FMC) based transcoding, as well as motion computation bypass (MCB) transcoding. We encoded a very high-resolution full motion MPEG-2 704x480 video stream (sequence "Basketball") with

original encoding at 10Mbps rate. In both cases the generated bit-rate successfully followed the target giving a compression ratio of about 1/6th – 1/7th times. (This scalability range is quite large compared to about 1/1.25 reduction limit achievable by network layer approach). We observed similar perfect feedback control for other video streams as well.

Fig-4(b) shows the corresponding picture quality distortion. The distortion is however, dependent on the video content. It shows the SNR for three different sequences "Shamu", "Mike Talking", and "Basketball". These three sequences have different levels of motion in them. The SNR varies between the FMC and MCB modes.

An important trend to note in these graphs is that at around 1/6-1/7th compression the quality dropped sharply, at the same time the rate of bit-rate reduction also slowed down. This slowdown indicates the limits that a transport protocol aware scheme also eventually faces.

The next level of down scalability requires most involved technique. One example is content aware transcoding. The frame dump taken from our transcoder shown in Fig-5(b). For comparison in Fig-5(a) we show a frame with quantization only scheme. These respectively show the difference created by a content unaware and content aware transcoding scheme. Both the sample frames[3] have same number of bits. But the frame on the right has higher perceptual quality. A careful inspection will reveal that *region of focus* (ROF) in the right has much higher detail than the one in the left. This has been achieved by the object detection technique, which was working in the transcoder. The region-wise bit-distributions are different (area SNR shown in Fig-6(a) and Fig-6(b)). The object-based transcoder took away more bits from the background but re-allocated them in the ROF. The transcoder we have developed is capable of varying both spatial and temporal quality of objects among the various ROFs in the volume video data space.

[3] On screen more details appears in the center. Compare the face, the white jaw line of the whale, and the tone of the water.

5.2 Computational Performance of Active Channel

Fig-7 shows the frame-rate observed in their sample run on a small uncontrolled (with background computational and communication load) active network consisting of 5 active routers (with capacity ranging from 400 MhZ ~ 1.5 GhZ P4 processors, and the interconnects were 10/100 Ethernets with uncontrolled cross traffic). We let the system auto deploy itself and find optimum mapping. Figure-7 plots the frame/ second statistics recorded at the GOP-MUX unit. It plots the performance for 16x120, 320x240 and 704x480 frame sizes streams.

The computation load heavily depends on the number of macro-blocks or frame size. Based on the frame size the frame transcoding rate varied from 30-5 frames/second.

The adaptive behavior is noticeable at the step

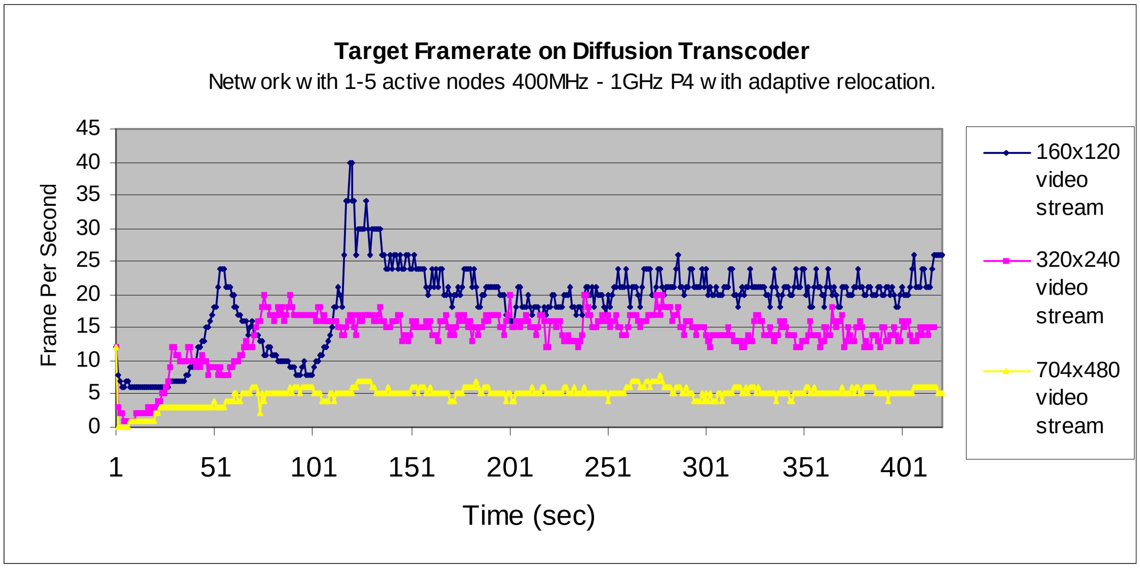

Fig-7 The adaptive growth of frame rate based on the GOP arrival times recorded at the GOP-MUX unit. The adaptive behavior is noticeable at the step like increments at the very beginning. Initially the channel used only one active node. The single node was unable to sustain the target rate, it auto-deployed additional active nodes.

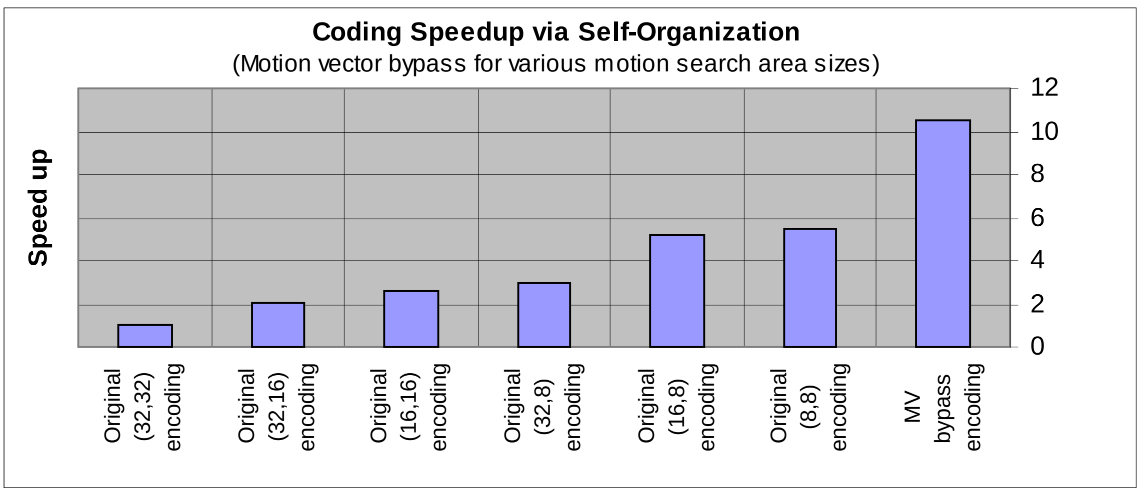

Fig-8 The speedup due to motion vector computation bypass. The actual speedup also depends on the motion vector search space.

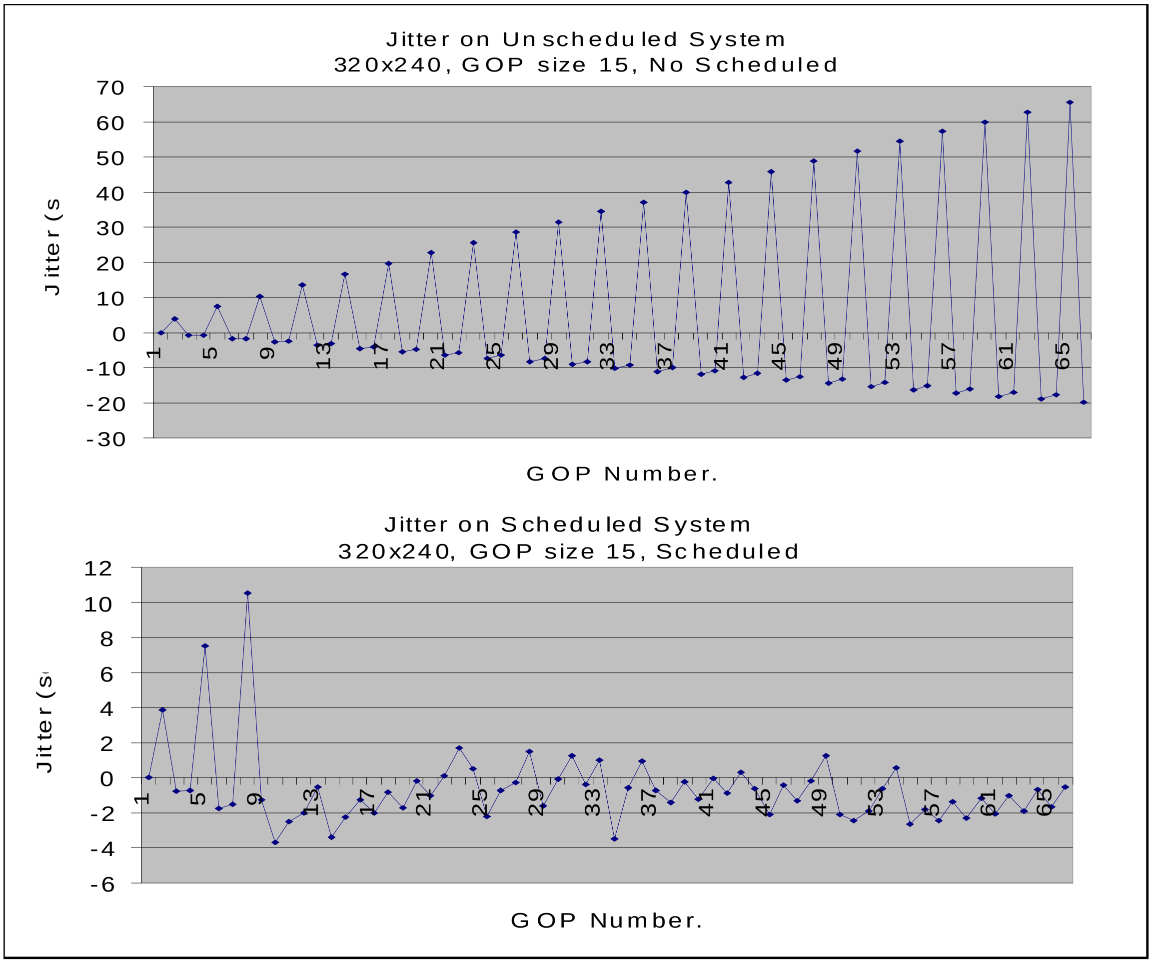

Fig-9. (a) The top graph shows the performance without jitter control scheduling. It uses a simple round robin assignment. The high jitter is due to the dominance of the low performance encoder(s). Also, the effects are accumulated as time passed. (b) The bottom graph sows the performance of the jitter controlled scheduler.

like increments at the very beginning. Initially the channel used only one active node. As the single node was unable to sustain the target rate, it auto-deployed additional active nodes. For example for 704x480 video the second and the third nodes were deployed some time before 20th and 60th seconds respectively. These delays represent the full feedback and effectuation delays It include (i) the time to detect insufficiency, (ii) the time for stream auto deployment, and (iii) the time it takes the new results to appear at the MUX. As evident from the jumps only three active paths were available. This is dependent on the underlying network configuration. The above results have been obtained from a transcoder running in full motion computation (FMC) mode. Further acceleration is achievable if motion vector computation bypass (MCB) mode is selected. The actual speedup however, is quite complex by the very nature of the paradigm. It will depend on the cost of full motion search which is also configurable, and the ability of computational paths (not only the computing power but also the required bandwidth). In figure-8 we show the speedup achieved on a single node MCB with respect to 32x32 motion search FMC, which shows another 2-10 times potential speedup. In the self organization process, based on the processing power can reduce the search size and eventually can move to MCB model

Another important concern was the jitter. For active channel and diffusion path computation jitter takes a full new meaning. The predominant source of jitter here is variation in the computational path delays, rather than the buffer of encoding delay. When jobs were scheduled in simple round robin the jitter grew enormously.

We added a new jitter control algorithm. Fig-9 shows the jitter. It plots the deviation of actual GOP arrival from its expected arrival time at the GOP-MUX (negative jitter means it arrived before expected time and thus it is harmless). The top graph shows the performance without jitter control scheduling. It uses a simple round robin assignment. The high jitter is due to the dominance of the low performance encoder(s). Also, it can be noted that the effect accumulated as time passed. The bottom graph shows the performance improvement due to the jitter controlled scheduler. It makes decision based on a path delay and variation of delay. At the beginning it has almost same delay as non-jitter controlled one, but after some initial delay, the scheduler adapted with difference of path delay. So it dramatically reduces the jitter due to path variation.

6. Discussions

In this paper we have described the automorphic stream in a single sender and a single receiver context. We have also designed a multiple end-point streaming system. This required adding a stream replication component. Here the downgraded video is replicated into multiple ports. The GOP-MUX unit is capable of adding a client at run-time. A new client can join the streaming session from the beginning of next GOP. Interestingly, the programmability in GOP-MUX localizes the sink-group management task, and is potentially scalable. The details of the component deployment mechanism can be found in a sister paper and in paper [14].

Adaptation is a complex process. In this research we have focused on resource adaptive network based systems to function in extreme environments. Success in adaptation comes from cognizance. In this transcoder we have selected techniques to demonstrate how the knowledge about the network, the knowledge about the content protocol and the knowledge about the content itself can be increasingly infused. The efficacy of adaptation increases at each successive level. Each level involves increasingly deeper domain knowledge. Scalable streaming will be increasingly important in the internet congestion control with the increase of video traffic in it. It is highly unlikely that required wide range rate scalability for video (or similar high volume traffics) can be achieved with the current network level only knowledge and schemes.

The case study presented here in the form of novel automorphic video stream represents an application with significant software and systems engineering complexity[4]. We have also crafted into it a carefully selected set of novel adaptive behavior. It also involves high volume data rate with temporal quality constraints.

Its rich capabilities also make it an arch type of many other net centric applications. XML/WAP Transcoder, Filter, or Application Servers in *content services networking* (CSN) systems will be examples of similar networked transcoding systems. These systems will involve similar mechanics for deep domain knowledge infusion at network points. The scalable multi-path flow feature will be common with the emerging application level adaptive routing based overlay networks, multimedia collaboration applications, CSN distribution back-end, and similar systems. The general model of network adaptation points (NAP) shown in figure-2 will appear for many adaptive applications.

An important issue in adaptive systems is the feedback response time. Lower level (kernel/ network) implementation of few of its functionalities will contribute quantitatively by speeding it up. It seems perceptual data communication may tolerate delays close to 1/10th to 1 second. However, for more tightly coupled adaptive loops (such as machine control), the delay have to be significantly reduced.

There is little support for cognizant adaptation in today's Internet infrastructure or in the deployed protocols. Today media sites pre-encode and store multiple copies of the same media one for each popular rate class (LAN, DSL/Cable, 56K, 28.8K etc.) [13,1,3,4]. The idiosyncrasy of the whole arrangement is that we expect the end-user to provide the information about the bandwidth. The end-user seems to be the last person to answer such a pure network parameter! Within the current network software/firmware infrastructure there is no easy means for such common sense intelligence, so we simply pass on

[4] By some account MPEG-2 is considered among the most complex protocol ever designed [3,4]. It has been designed to carry digital video compressed with major compression techniques with high coding efficiency, and in wide rate scale.

this question to the poor end-user! Perhaps infrastructure for the cognizant adaptation has to be built from the core if we expect to build applications, which are truly user friendly.

Very little intelligence can be squeezed out if the information base on which a system operates is dry. The IPV6 extension headers will allow space for infusion of more information into network for realizing some adaptive behavior. What is in short supply now is the flexibility of acting on it in the right place in the right time. Without some network embedded programmability their use will remain limited. It seems smart programmable network may have a destined role to play here.

The work is currently being funded by the DARPA Research Grant F30602-99-1-0515 under its Active Network initiative.

7. References

[1] Amir, Elan, Steven McCanne, and Randy Katz, Receiver-driven Bandwidth Adaptation for Light-weight Sessions, Proceedings of ACM Multimedia '97, Seattle, WA, Nov 1997.

[2] Bhattacharjee, S., Kenneth L. Calvert and Ellen W. Zegura. An Architecture for Active Networking. High Performance Networking' 97 ,White Plains, NY, April 1997. [http://www.cc.gatech.edu/projects/ canes/papers/anarch.ps.gz, October 98]

[3] Fluckiger, F, Understanding Networked Multimedia Applications and Technology, Prentice Hall, UK, 1995.

[4] Haskell B. G., Atul Puri and Arun Netravali, Digital Video: An Introduction to MPEG-2, Chapman and Hall, NY, 1997.

[5] Information Technology- Generic Coding of Moving Pictures and Associated Audio Information: Video,ISO/IEC International Standard 13818-2, June 1996.

[6] Javed I. Khan, S. S. Yang, Medianet Active Switch Architecture, Technical Report: 2000-01-02, Kent State University, [available at URL http://medianet.kent. edu/ technicalreports.html, also mirrored at http:// bristi.facnet.mcs.kent.edu/ medianet]

[7] Keesman, Gertjan; Hellinghuizen, Robert; Hoeksema, Fokke; Heideman, Geert, Transcoding of MPEG bitstreams Signal Processing: Image Communication, Volume: 8, Issue: 6, pp. 481-500, September 1996,

[8] Javed I. Khan, Motion Vector Prediction in Interactive 3D Video Stream, Proceedings of the World Congress on Advanced IT Tools, IFIP '96 IT, Canberra, Sept 96 , pp533-539.

[9] Tennenhouse, D. L., J. Smith, D. Sincoskie, D. Wetherall & G. Minden., "A Survey of Active Network Research", IEEE Communications Magazine, Vol. 35, No. 1, Jan 97, pp 80-86

[10] Wetherall, Guttag, Tennenhouse, "ANTS: A Tool kit for Building and Dynamically Deploying Network Protocols", IEEE OPENARCH'98, San Francisco, April 1998. Available at: http://www.tns.lcs.mit.edu/publications/openarch98.html.

[11] Youn, J, M.T. Sun, and J. Xin, "Video Transcoder Architectures for Bit Rate Scaling of H.263 Bit Streams," ACM Multimedia 1999', Nov., 1999. pp243-250.

[12] Jill M. Boyce and Robert D. Gaglianello, Packet loss effects on MPEG video sent over the public Internet; Proceedings of the 6th ACM international conference on Multimedia , 1998, Pages 181 - 190

[13] White Paper, Delivering RealAudio® or RealVideo® from a Web Server, RealNetworks Technical Blueprint Series, 1998 [URL:http://service.real.com/help/ library/blueprints/ hosthtml/webhost.htm Last Download: Sept 20, 2000]

[14] Javed I. Khan, & S. S. Yang, Made-To-Order Custom Channels for Netcentric Applications over Active Network, Proc. Of the Internationaal Conf. On Internet and Multimedia Systems and Applications, IMSA 2000, Nov 2000, Las Vegas, pp22-26.

[15] Katie Hafner and M. Iyon, Where Wizards Stay up Late: the Origins of the Internet, Simon and Schuster, New York,

1996.

[16] Overview of the MPEG-4 Standard, ISO/IEC JTC1/SC29/WG11 N3444, Geneva – May/June 2000, [Also available from http://www.cselt.it/mpeg/standards /mpeg-4/mpeg-4.htm]

[17] Javed I. Khan, Darsan Patel, Wansik Oh, Seung-su Yang, Oleg Komogortsev, and Qiong Gu, Architectural Overview of Motion Vector Reuse Mechanism in MPEG-2 Transcoding, Technical Report TR2001-01-01, Kent State University, [available at URL http://medianet.kent.edu/ technicalreports.html, also mirrored at http:// bristi.facnet.mcs.kent.edu /medianet, January, 2001]

[18] Jill M. Boyce and Robert D. Gaglianello, Packet loss effects on MPEG video sent over the public Internet; Proceedings of the 6th ACM international conference on Multimedia , 1998, Pages 181 – 190

[19] Gary Stix, The Triumph of the Light, Scientific American, January 2001, pp31-35 [HTTP:// http://www.sciam.com/ 2001/ 0101issue/ 0101stix.html]

[20] [Peter Forman and Robert W. Saint, Creating Convergence, Scientific American, November 2000, pp.10-15[HTTP://http://www.sciam.com/2001/0101issue/0101stix.html]

[21] Javed I. Khan and Asrar Haque, An Active Programmable Communication Harness for Measurement of Composite Network States, Submitted to IEEE International Conference on Networking, ICN' 2001, pp628-638.

[22] Xuebin Xu, "A Survey of Ideas in Programmable Networks", Technical Report: 2002-01-01, Kent State University, [available at URL http://medianet.kent.edu/ technicalreports.html, also mirrored at http:// bristi.facnet.mcs.kent.edu/ medianet]

[23] Andrew T.Campbell, etc., "A Survey of Programmable Networks" Computer Communication Review, vol.29, no.2, pp.7-23, Apr.1999.

[24] 11. Thomas M.Chen, etc., "Active and Programmable Networks", guest editorial for Special Issue: Active and Programmable Networks, Network Interactive, May 1998.