Multimedia Systems
DOI 10.1007/s00530-008-0124-2

REGULAR PAPER

A biologically inspired multimedia content repurposing system in heterogeneous environments

M. Shamim Hossain · Abdulmotaleb El Saddik

Abstract The application of biologically inspired algorithms to multimedia repurposing systems in heterogeneous network environments is gaining importance due to its ability to adapt and customize multimedia content to frequently changing network environments. This paper presents a biologically inspired proxy-based multimedia content adaptation system, which is used to repurpose multimedia content dynamically for the transmission over heterogeneous networks and end devices. We use a series of repurposing proxies' services in a chain fashion between the sender and multiple end devices. In order to find the appropriate chain of repurposing services that satisfy the Quality of Experience (QoE) requirements, the proposed service selection algorithm uses the ant colony metaphor. During the communication session, the algorithm uses biological foraging behavior inspired from ant agents to find optimal service paths between the media sender and the destination. Experimental results demonstrate that the proposed algorithm provides significant performance gain over traditional, state of the art selection algorithms and saves the average delay.

Keywords Multimedia content repurposing · Service selection · Biologically inspired algorithm

1 Introduction

The need for a multimedia content repurposing service is ever increasing due to the rapidly growing diversity of resource-hungry heterogeneous hand-held devices (PDA, Laptop, Cell Phone, etc.) and their heterogeneous network connections. Hand-held devices differ with respect to multimedia-related capabilities such as display size, color depth, processing power, memory capacity, battery life, and the ability to handle different multimedia formats (H.263, H.264, MPEG-4, etc.), while their network connections have different Quality of Service (QoS) support, bandwidth limitations and variations. However, most of the multimedia content designed for desktop computers and high-speed networks are not suitable for wireless hand-held devices with limited display capabilities, processing power, and network bandwidth [1]. As a result, a multimedia content repurposing service is used to eliminate the mismatch between the rich multimedia content and the limited wireless terminal capabilities.

A multimedia repurposing service is a self-contained multimedia application that provides transcoding or repurposing functionality, where it receives the input in a given format and produces an output in a different media format. Figure 1 shows an example, where a user sends a request for content to a media server in a multimedia system that supports content repurposing. In the system, several repurposing services $\{R_1, R_2 \dots R_N\}$ are distributed in several proxies over networks with each repurposing service having different capabilities. Now, the system has to decide to apply appropriate repurposing services on the content before it is delivered to the user, according to his satisfaction, by considering the resource constraints. Selecting the appropriate repurposing service and the optimal repurposing service path based on QoE [9] is a challenge.

To overcome the challenge, modified traditional shortest path selection algorithms based on Dijkstra [2] or Bellmanford [3] are used. Traditional algorithms choose a path with the minimum cost, which could produce a bottleneck. In addition, under highly dynamic traffic and

M. S. Hossain (✉) · A. El Saddik
Multimedia Communications Research Laboratory (MCRLab), SITE, University of Ottawa, Ottawa ON K1N 6N5, Canada
e-mail: shossain@site.uottawa; shamim@mcrlab.uottawa.ca

A. El Saddik
e-mail: abed@mcrlab.uottawa.ca

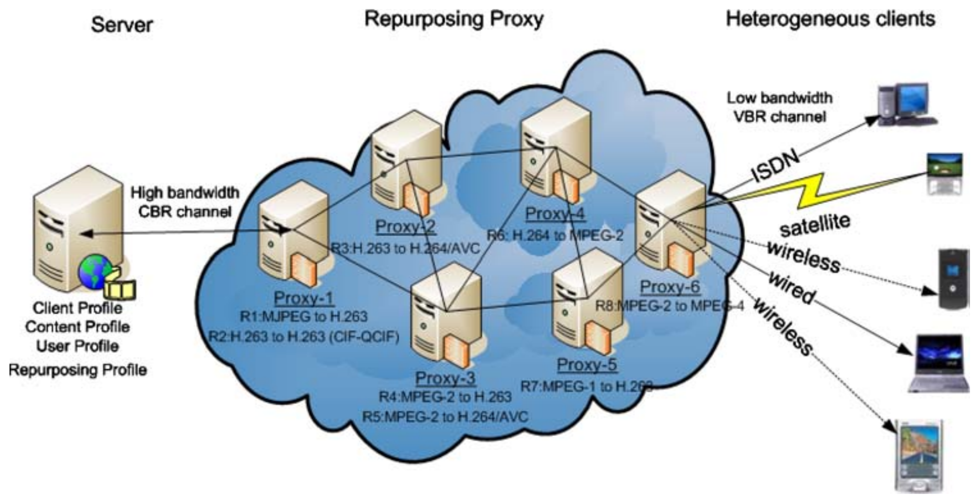

Fig. 1 Multimedia repurposing service selection scenario

heterogeneous network conditions, the performance of a traditional algorithm can seriously be degraded and subjected to error or loss. This causes high user latency and repurposing delays. Beside these, the traditional methods do not have enough flexibility to satisfy new service demands, end user's preferences and QoS for multimedia transmission over a highly dynamic heterogeneous wireless network environment.

One way to cope with the above-mentioned problems with traditional algorithms is to draw inspiration from biological systems. Biological systems can effectively organize/manage large numbers of unreliable units, which can be referred to as nodes or multimedia services in network environments. Biological systems are robust to failure or loss [4]. It has been a long research challenge to characterize the ideas that make biological systems work and apply them to multimedia systems in a heterogeneous network environment. In addition, the biologically inspired system is robust to error and is able to survive and evolve in a highly dynamic wireless network environment [4].

In this paper, we propose and design a repurposing service selection algorithm, which is adopted from AntNet [5]. AntNet is based on the Ant Colony Optimization (ACO) meta-heuristics [6] inspired by the collective foraging behavior of ants that use a chemical substance called "pheromone" for indirect communication. These pheromone trails deposited by ants, attract other ants to find the shortest path from a starting point to a destination. Based on the problem, an ant is given a starting point and moves through a series of intermediary neighboring states to find the shortest path, which is considered as the optimal solution, by applying a stochastic decision policy.

The remainder of the paper is organized as follows. In Sect. 2 we briefly describe the system with a traditional selection algorithm. Section 3 introduces our new proposed Biologically Inspired Service Selection (BioReSS) algorithm. The algorithm can find an optimal repurposing path, which maximizes the user's QoE.We also define and prove that the optimal service selection problem as Nondeterministic Polynomial-time hard (NP-hard). In Sect. 4, we discuss the implementation of BioReSS. Experimental results and performance evaluation of the proposed algorithm are discussed in Sect. 5. Section 6 discusses related works and Sect. 7 concludes the paper and gives some possible future directions.

2 System with a traditional selection algorithm

In the repurposing system with a Traditional Selection Algorithm, the initial connection between the client and the server is established using the Session Initiation Protocol (SIP) [7]. Initially, the server sends a SIP "invite" message to the client containing the available ports and the IP of the server. The client replies with a 200 OK message to which it adds its profile including the device's capabilities, the IP-address of the device and the supported codecs. Upon connection with the server, the server stores the client's profile. The server then connects to each of its neighboring proxies. From each of the neighboring proxies, the server requests the profile of the proxy that includes one or more connected neighboring proxies and the repurposing services that are running on each neighboring proxy. Once all the information from all the proxies is received, the server generates a repurposing graph starting with the services that the server provides and expands the graph until all the proxies, and the required repurposing services have been taken into consideration and the client has been reached. Then, the graph is simplified in order to ensure that no infinite loop exists and there is no backtracking. The graph is simplified in such a way that all paths lead to the receiver and there are no extra edges between the neighboring repurposing services. After the graph simplification, the server discovers the repurposing path that provides the best quality (user satisfaction) to the receiver. The path discovery uses the proposed satisfaction metric in Eq. (1) and the traditional selection algorithm presented in [10].

Once the server creates the best repurposing path, it starts streaming the captured and/or pre-encoded visual content using the Real Time Protocol (RTP) [8]. The server then sends the path information and the stream to the first proxy in the chain. The proxy, upon receiving both path and stream, creates one or more repurposing services instantly. A stream can pass through multiple proxies and repurposing services and, if necessary, the stream can pass through the same proxy twice using two different repurposing services. However, it cannot use the same proxy twice in a row in the repurposing chain. The proxy then sends the rest of the path, as well as its own extracted information, to the next proxy in the chain such that the output formats of the previous repurposing service matches the input format of the next repurposing service in the chain. The last proxy in the chain simply forwards the results of its repurposed content to the client. Once the streaming through a repurposing service stops, the reception port of the proxy is freed and the repurposing object is destroyed.

The aforementioned repurposing system with traditional algorithms depends on the global exchange of information among repurposing service nodes for path or service selection and it become unfeasible due to dynamic changes of topology or unknown destination. It cannot respond immediately to network changes or repurposing service node failure. In order to overcome some of these challenges, systems with bio-inspired selection are getting research attention. In the following section, we are going to describe the proposed system with the Biologically Inspired Service Selection (BioReSS) approach. BioReSS depends on the local exchange of information among repurposing service nodes, where each node autonomously sends ants through the network for the best repurposing path or service selection. Under an increased network load, BioReSS has better performance than the traditional approach. In contrast, under a decreased load both the traditional and BioReSS approaches have similar performances.

3 Biologically-inspired service selection (BioReSS) algorithm

Biologically inspired routing is now a new area of research as they can solve the complex problem of routing through simple agents (ants) that are sent through networks to gather the required routing information. This algorithm inspired from a biological "foraging" concept, which provides an efficient utilization of network resources in response to constant changes of networks. We first describe the AntNet Algorithm 2.1, and then we formalize the problem to be addressed in Sect. 2.2. Finally, in Sect. 2.3 we discuss our proposed BioReSS algorithm and the changes we made in order to adapt the AntNet into repurposing service selection.

3.1 AntNet algorithm

AntNet is based on the Ant Colony Optimization (ACO) meta-heuristics [6] inspired by the collective foraging behavior of ants that use a chemical substance called "pheromone" for indirect communication. These pheromone trails deposited by ants, attract other ants to find the shortest path from the starting point to a destination. Based on a specific problem, an ant is given a starting point and moves through a series of intermediary neighboring states to find the shortest path (optimal solution) by applying a stochastic decision policy. In AntNet [5], the data network is mapped on a graph with N nodes, where s and d are the source and destination nodes, respectively. Each node has a routing table that stores information about the outgoing links and their amount of pheromones. The routing tables for each reachable destination are initialized with a uniform distribution of probability (equal value).

As can be seen in Fig. 2, we have forward and backward ants. At regular intervals, each node s launches a forward ant $F_{s\to d}$ to a randomly selected destination d stochastically. After that, a forward ant $F_{s\to d}$ applies the transition rule to choose the next node. This transition rule is based on the traffic load (link cost) and the amount of pheromones. While moving each forward ant $F_{s\to d}$, collects information about the state of the network, which is later used by backward ants $B_{s\to d}$, to update the routing tables along the path followed. In order to avoid the cycle, the forward ant is forced to return to an already visited node. Upon successfully reaching the destination d, it generates a backward ant $B_{s\to d}$ transfers all the information to the backward ant and then dies. The backward ant $B_{s\to d}$ returns to the source node s using the same path that was used by $F_{s\to d}$. The backward ant $B_{s\to d}$ updates the corresponding routing table and the traffic model of each visited node. On reaching the source node s, it dies.

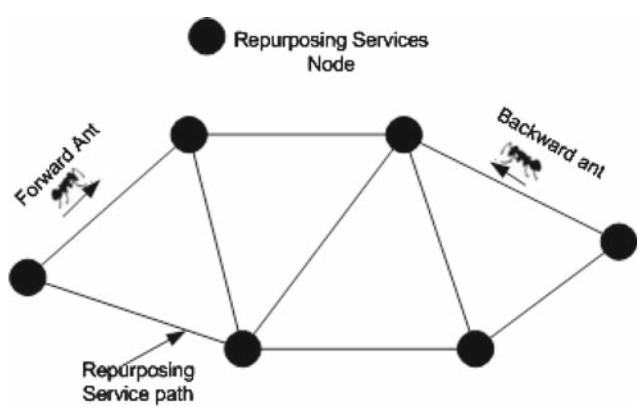

Fig. 2 Repurposing service graph with a forward and backward ant

3.2 Formalizing the research problem

To find or select an optimal repurposing path between the multimedia sender and receiver in the Ant system, we consider directed acyclic graphs (DAGs), as presented in Fig. 2, $G_s = (V, E)$ where V represents a set of X proxy nodes $V_j = \{v_1, v_2 \dots v_X\}$, $1 \leq j \leq X$ and E represents a set of Y links, edges, or service paths $E_l = \{e_1 \rightarrow e_2, \dots . e_2 \rightarrow e_Y\}$, $1 \leq l \leq Y$. Each node V provides a set of N repurposing services $R_v = \{r_1, r_2 \dots r_N\}$, $1 \leq v \leq N$. The cost associated with each link of the repurposing service network is expressed as a positive weight $W : E \rightarrow Q_s^+$. The cost or weight refers to the QoE [9] (Q_e^+) for the user, which is a function of the Quality of Service (Q_s) and the user's satisfaction, $Q_e^+ \rightarrow \{Q_s, S_i\}$. $Q_s \rightarrow \{q_1, q_2 \dots q_3\}$ represents the set of (Q_s) criteria related to network resource level variables such as delay, jitter, bandwidth, etc. The user's satisfaction (S_i) is based on the multimedia application level variables such as the frame rate, resolution, and coding distortion (quality), as desired by the user. The satisfaction or preference of a user with each application level's variables is expressed as a component satisfaction function $s_i(y_i)$, whose value is between 0 and 1. In this function y_i is a multimedia application level variable. For more than one multimedia application parameter, or variable, the overall satisfaction [10] of a user is determined as a combination function of the individual component satisfaction s_i, presented in Eq. (1) below:

$$S_{\text{comb}} = f_{\text{comb}}(s_i) = f_{\text{comb}}(s_1, s_2 \dots s_m) = \frac{m}{\sum_m^{i=1} 1/S_i}, \quad 0 \leq S_i \leq 1 \tag{1}$$

Where, each satisfaction component $s_1, s_2 \dots, s_m$ is based on some parameters such as frame rate, resolution, signal to noise ratio (SNR) quality, etc.

Now, we want to find an optimal service path with the maximum user's satisfaction while satisfying the QoS constraints of the networks and paths. Finding the optimal service path and server selection can be regarded as a restricted shortest path problem or a bottleneck Traveling Salesman Problem. These problems have already been proven as NP-hard [11].

We apply the ant colony metaphor for selecting the best repurposing service route or path, which satisfies both the requirements of the users and resources. In order to select a repurposing service among proxies, we adopt AntNet algorithms, where the ant controller packet is responsible for sampling and generating the repurposing path among the repurposing services running on intermediate proxies.

The basic idea of repurposing is as follows: The sender sends a stream to a proxy, which it determines to be the best, together with information about the destination. The proxy receives the stream and determines the next node on the path based on the routing table, as well as what the next node can accept as a format. It then creates a repurposing service to repurpose from the format it received to the format required by the next node. Next, the data coming from the repurposing service is streamed to the next node, together with the information about the destination. This is repeated until the receiver receives the required stream.

3.3 Adapting the ant metaphor to BioReSS for content repurposing

In our Biologically Inspired Repurposing Service Selection (BioReSS) System, we use the same transition rule, the pheromone-updating rule and the model data-updating rule as in [5]. However, we use a combination of QoE metrics (both the application level QoS, based on user's satisfaction, and the network resource level QoS) with the pheromone for selecting an optimal repurposing service path or optimal (one of the best) repurposing services. The QoE metric refers to the overall end-to-end service level performance from the user's perspective relative to their expectations and requirements about a multimedia service or application. The QoE (Q_e) is more crucial to consider than the QoS in selecting multimedia services. The QoE has a direct impact on how well the multimedia service fulfills the user's needs rather than the internal or implied impact of the QoS [12]. The QoS is a subset of the QoE and it may have an effect on the QoE. The QoE is a multi-constraint complex function, which may include the user's satisfaction/usability, response time, and multimedia quality parameter (e.g., frame rate, and QoS related parameter (e.g., delay, bandwidth, etc.). Considering the QoE (Q_e) was also mathematically defined in the previous section,let us see the changes and other related features of BioReSS.

In the forward phase, the ant stores the QoE, which is obtained from the service path or link. As mentioned before the QoE is calculated based on network level parameter (e.g., bandwidth) and the user satisfaction or application level parameter (e.g., frame rate). This is required to send the video via that link. The normalized value for QoE (Q_e) is based on the frame rate, where the maximum value of the frame rate is 30. Therefore, the formula of calculating local heuristic variables is in Eq. (2)and f is calculated using the following Eqs. (3) and (4):

$$l_n = \frac{(Q_e^+)}{(Q_{max})} = \frac{f}{30}, \quad 1 \leq f \leq 30 \tag{2}$$

$$\beta_\gamma = \{\xi \times \beta\}, \quad \xi = \sqrt{\alpha/\mu} \tag{3}$$

$$f_a = (\beta_a)/\beta_\gamma) \times f_{\text{max}} \tag{4}$$

Where, Table 1 represents the descriptions of symbols used in above equations.

In the backward phases, the routing table is updated by changing the value of the QoE that is obtained by sending the video/audio stream via the next node in the path towards a certain destination. In the BioReSS algorithm, it is the

Table 1 Description of Symbols

Symbol	Meaning
α	Resolution (e.g., CIF, QCIF) of the multimedia (e.g., video) stream
μ	Resolution of the stored video (e.g., H.263, H.264)
β	Bandwidth of the stored video stream
βa	Available bandwidth for the multimedia stream
βr	Bandwidth required for the multimedia (video) stream in order to get maximum user preferences (frame rate)
$f_{\max}$	Maximum frame rate for the multimedia stream
f_a	Available frame rate for the multimedia stream

responsibility of each proxy node to decide the next node on the path and what stream it should be repurposed to, in order to obtain best the QoE at the receiver. Furthermore, in the BioReSS algorithm, the graph of the network is generated as the ants are coming back. Each node knows to start with only its 1 hop neighbors. At each node, the average QoE, the best QoE and the variance of the QoE for each destination are maintained. As backward ants return, the routing table is extended with nodes that can be reached via multiple hops. The backward ants generate and reinforce information related to the optimal path. This information is stored in a local routing table on each proxy node. This information include among others, the format of the required media, QoE, the final destination as well as the next proxy node in the chain—if any—where the output stream should be sent in order for the repurposing process to be achieved.

4 Implementation of BioReSS algorithm key components

Figure 3 represents the class structure view for the proxy component that implements the Ant algorithm. In the following, we will briefly explain how these different classes interact with each other. However, the detailed discussion of the complete system design and architecture is beyond the scope of this paper.

First, each proxy in the system contains one single *AntController* object, which is responsible for initializing the ant algorithm by reading the XML configuration file as presented in Fig. 4. It is also responsible for managing the generation and reception of *Ant* objects. This is achieved by running two separate threads: the *AntGenerator* and the *AntReception* respectively. In addition, the *AntController* initializes the proxy with different properties, such as the video repurposing type, and the number of frames per second. It stores them in the *ModelData* object for later use. The *AntGenerator* thread creates a *ForwardAnt* object towards the destination node. To do so, the *ForwardAnt* decides based on the proposed selection criteria and the available neighbors through *TraversableNode*object. Then, the *AntGenerator* forces the *ForwardAnt* object to jump to that *TraversableNode*.

Upon arrival of the *ForwardAnt* at the *TraversableNode*, the *AntReception* reads the *ForwardAnt*object, stacks this node into the *TraversedNode* under the *ForwardAnt,* and examines the *ForwardAnt* destination information against the node information, which is retrieved from the *ModelData* object. This scenario is repeated at each node along the path toward the destination node. If the node is the final destination of the *ForwardAnt* object, then the *AntReception* at that node forces the *ForwardAnt* object to create a *BackwardAnt* object and forces it to jump back toward the *ForwardAnt* object originator node before it dies. The *BackwardAnt* utilizes most of the *ForwardAnt*'s collected information started from the source node. At each node along the path back to the source node, the *AntReception* repeats the same scenario with the *BackwardAnt as* it happens with the *ForwardAnt* object. Finally, when the *BackwardAnt* arrives at its source node, it reports different statistical information and stores it in the ModelData object before it dies.

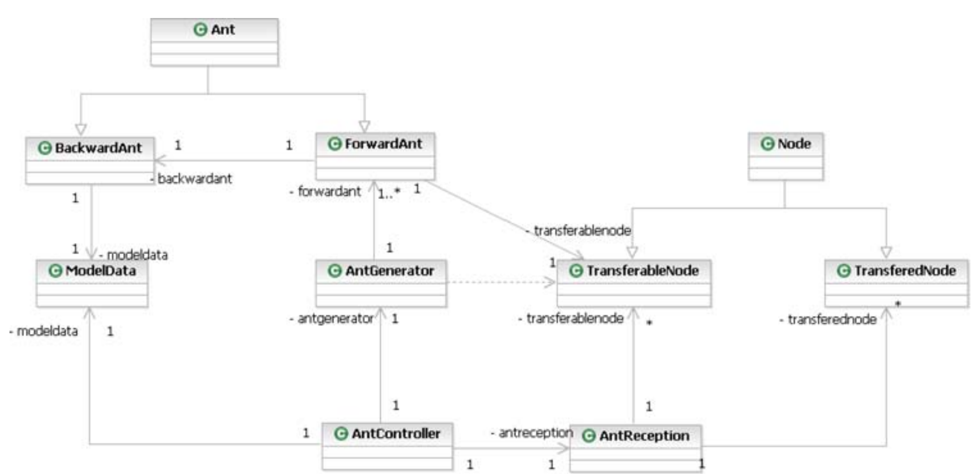

Fig. 3 A class diagram for the BioReSS algorithm implementation

Fig. 4 Example of a simple ant proxy XML configuration file

```
<?xml version="1.0" encoding="UTF-8"?>
<proxyNode xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xsi:noNamespaceSchemaLocation='file:/c:/MCR/proxyNodeDefinition.xsd'>
<RepurposeDirectory>
      c:/RepurposeProxy/AntTranscProxy1/proxy1Dir.xml
</RepurposeDirectory>
<IP>137.122.89.112</IP>
<pathPort>20204</pathPort>
<antPort>30204</antPort>
<neighborList>
      <neighbor neighborDir="h:/MCR/senderProfile.xml"
      IP="137.122.91.89" dataPort="22322" pathPort="20202"
      antPort="23000" bandwidth="500000" type="in"/>
      <neighbor neighborDir="h:/MCR/AERIC.xml" IP="137.122.90.167"
      dataPort="22232" pathPort="20204" bandwidth="500000" type="out"/>
      <neighbor neighborDir="h:/MCR/receivProxy.xml" IP="137.122.91.223"
      dataPort="22240" pathPort="23000" antPort="23000"
      bandwidth="813700" type="out"/>
</neighborList>
</proxyNode>
```

5 Performance evaluation and results

We first evaluate the BioReSS algorithm with the traditional service selection algorithm presented in [10]. For the traditional service selection algorithm, we implemented a dijkstra-based service selection algorithm. We then incorporate the BioReSS algorithm in a multimedia repurposing system to see how it performs at service selection and/or service path finding.

5.1 Comparison with the ttraditional service selection algorithm

In these experiments, we compared the throughput and routing overhead of BioReSS and traditional algorithm.

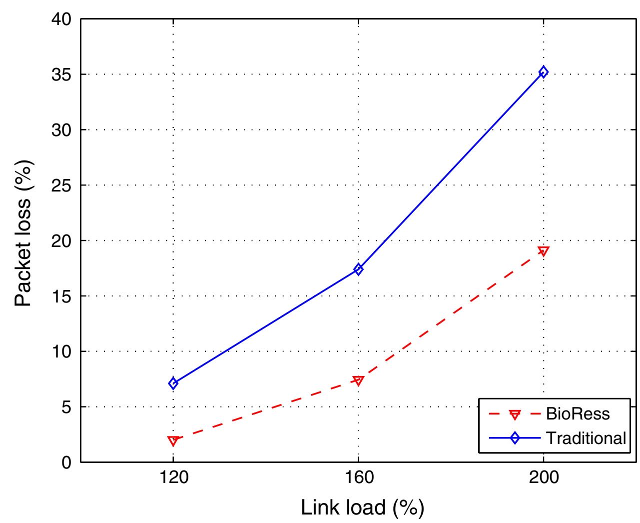

Fig. 5 Throughput comparison of traditional and BioReSS algorithms

5.1.1 Throughput comparison

In this comparison, we measured the number of lost packets under an increased network load. In BioReSS, the ants are sent via a socket through the use of Java Serialization, while the video/audio stream is sent via RTP/RTCP [8] through the use of Java Media Framework (JMF) [13]. Consequently, the ants are sent via high priority queues. As a result, fewer packets are dropped under the increased network load in BioReSS. This causes the increase in routing overhead in our proposed BioReSS over the traditional shortest path algorithm. As plotted in Fig. 5, BioReSS shows higher throughput than the traditional shortest path selection algorithm. This is because the proposed BioReSS algorithm forwards extra packets to the destination while these extra packets do not exist in the traditional shortest path algorithm. In BioReSS, the whole network is used efficiently, while in the traditional approach it is not used efficiently when the network links are under an increased load. In that case, some links are not used at all.

5.1.2 Routing overhead comparison

In this comparison, we compare the overhead generated by routing packets of BioReSS with that of the overhead of the traditional. For each algorithm, the overhead is measured as the ratio of the bandwidth occupied by routing packets to the total available bandwidth in the network. For this experiment, we use a network topology similar to that of Fig. 2, which has 6 nodes and 9 bi-directional links. The link bandwidth is around 1.0 Mb/s and link delays range from 5 to 10 ms. For the overhead comparison, the simulation is conducted in Java. The simulation runs for 200 s with the initial 45 s for initializing routing tables for network topology. We consider the packet size of 256 bytes and the mean packet inter-arrival time of 0.001 s, therefore, the transfer rate is approximately 1.04 Mb/s ($128 \times 8 \times 1000$). All reported data in the graph is averaged 5 times.

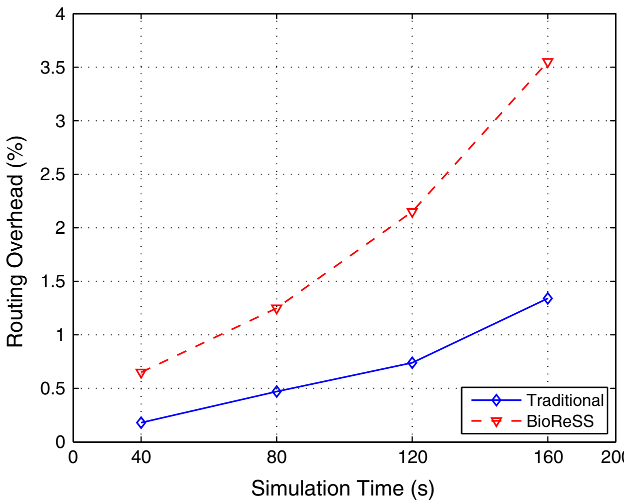

Fig. 6 Routing overhead comparison of traditional and BioReSS algorithms

Figure 6 depicts the overhead comparison generated by routing packets for both algorithms. As in BioReSS, ant packets are required to be sent periodically to monitor and collect the state of the repurposing paths and networks. The routing overhead of BioReSS is significantly higher than that of the traditional. The periodic updates of routing packets cause the increased overhead in BioReSS. In addition, network topology changes also cause the increased overhead in BioReSS.

One way to reduce the overhead generated by an ant packet is to adapt the approach proposed by [14], where the approach does not maintain routes to all possible destinations, but sets up routes only on demand. This causes less number of routing packets and control packets in the network, which eventually reduces the overhead generated by the packets. The service paths are setup in the same way as BioReSS. During the data session, the paths are monitored, maintained and improved using proactive forward ants [14]. In this way, the BioReSS can more quickly respond to link failures or other dynamic changes.

Even though the routing overhead of the BioReSS is significantly higher that of the traditional it does not affect our proposed system as (a) it is negligible [15] compared to the resource consumption compensated by the added performance than traditional (b) the streaming subsystem works independently from the rest of the multimedia conferencing application system. When a user wants to stream a video during a conferencing session, it is assumed that the BioReSS program has been already working in the background and generating the network topology, structure and values. Thus, extra overhead does not affect the proposed repurposing system.

Finally, BioReSS is more robust than the traditional, even at the time of service node or service link failure, throughput does not decay completely, and it can recover from that state which consequently gives a high throughput. In the case of dynamic changes in the heterogeneous network (e.g., wireless, mobile) where a node can join and leave the network, it can handle it robustly. However, for the traditional approach, the affect is nearly opposite.

5.2 Experiments with multimedia systems

We conducted experiments to study how our biologically inspired repurposing service selection algorithm behaves for real time multimedia applications. The results of the experiment show the effectiveness of the proposed BioReSS algorithm for selecting repurposing services in order to repurpose multimedia content. In order to do the tests we deployed the proposed algorithm in a multimedia conferencing service environment. This application periodically scans for available multimedia repurposing services and determines the best available service, with the help of the proposed service selection algorithm, in order to render the multimedia stream to different clients based on the user's satisfaction. After running the application, we measured the quality of the repurposed content. For the measurement, we used a widely accepted objective measure of visual quality metric called the Peak Signal to-Noise-Ratio (PSNR) defined in Eq. (5). Where, MSE is the mean square error between the original content and the reconstructed visual content.

$$\text{PSNR} = 10 \log_{10} \frac{255^2}{\text{MSE}} dB \quad (5)$$

The tests are conducted 5 times by using 15 repurposing services (node). The repurposing delay is measured as the delay in sending, repurposing, and receiving the stream from the media sender to the receiver. The results are compared to the previously developed system [10] that uses a traditional repurposing service selection strategy. As shown in Fig. 7, the BioReSS algorithm outperforms the traditional selection algorithm in terms of average delay.

As shown in Table 2, the path generation time for the BioReSS is virtually 0, because the next node (sender or proxy) calculation is finding the neighbor with the best QoE from a hash table stored locally. Consequently, we find that the quality of the repurposed content in both cases is almost the same (e.g., 32 dB). In order to test this quality, the system takes a live stream of motion JPEG [16] at 30 fps and repurposes it to H.263 [17] at a frame rate of 15 fps.

Table 2 Performance comparison of the system

	BioReSS	Traditional
Average path generation time	Virtually 0 ms	209.1 ms
Repurposed content quality	32.05 dB	32.06 dB

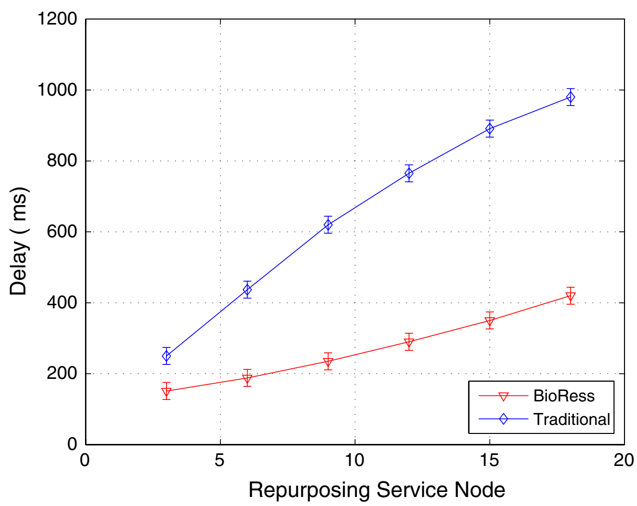

Fig. 7 The repurposing delay in two algorithms after having ported on a multimedia application

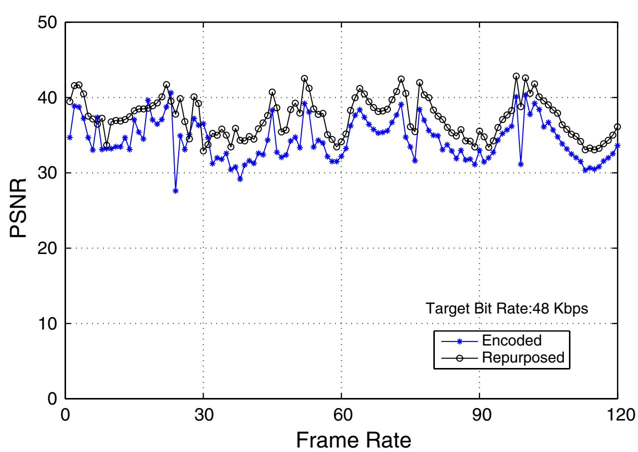

Fig. 8 Quality comparisons of the repurposed stream and the encoded stream at 48 kbps

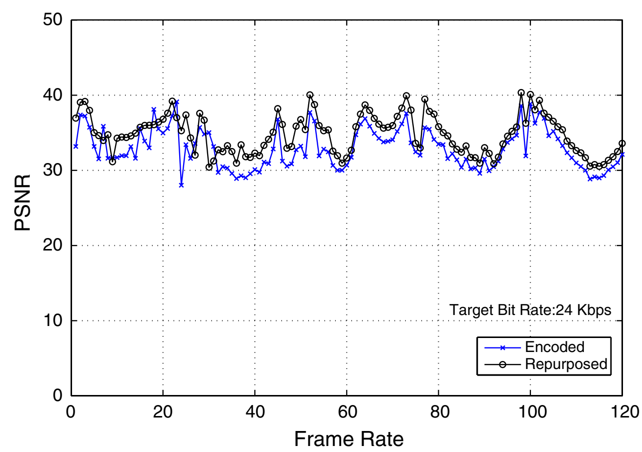

Fig. 9 Quality comparisons of the repurposed stream and the encoded stream at 24 kbps

We reported some significant subsets of the results of the experiments conducted on our multimedia content repurposing system after deploying the BioReSS. One of the repurposing services that we developed was able to repurpose video content from Motion JPEG (MJPEG) to H.263 at different target channel bandwidths. As mentioned before, we measured video quality by calculating and comparing the PSNR of repurposed video content and encoded content. As shown in Fig. 8 and Fig. 9, the video content is repurposed from the MJPEG to the H.263 for the low bit rate target channels of 48 and 24 kbps.

There is a sudden drop in both graphs in Figs. 8 and 9 for the encoded data series, which is due to the changing of the scene. However, even when the scene changes happen at the 24th and the 99th frame, the quality of the repurposed frame is higher than that of the encoded stream because of the repurposing. There is a quality gain of 2 db after repurposing the video content into H.263. This quality gain is due to the repurposing of the multimedia stream, where bandwidth is controlled.

6 Related work

Bringing the idea of a biological metaphor into multimedia systems or the adaptation of network resources is a new area of research. According to the best of our knowledge, only a few researchers use biologically inspired algorithms or mechanisms in distributed and multimedia systems. Babaolgu [18] presents the applicability of biologically inspired design patterns to distributed computing. The BISON [19] (Biology-Inspired techniques for Self-Organization in dynamic Networks) project uses the ideas derived from biological systems to enable the design and development of algorithms for deployment and control in highly dynamic heterogeneous network environments such as P2P, MANET, Wireless Multi-hop ad hoc, etc.

Suzuki [20] describes and proposes a biologically inspired adaptation mechanism for the adaptation of network applications in the NetSphere architecture, iNet [21]. In this architecture, the biological system is inspired from a Bee colony metaphor, where bee agents mimic functional services related to network applications in order to autonomously (a) monitor their surrounding environmental conditions (e.g., traffic) and (b) accommodate their behavior and customize it for the current environmental conditions. However, our work is different from [20] in selecting multimedia services and using it in a multimedia system. Our work is inspired from an Ant colony metaphor while, the work in [20], is inspired from a Bee colony metaphor [22].

Beside those of the above, to the best our knowledge, there is very little work related to the application of ACO [6] in multimedia content repurposing other than some applications in image processing, which are as follows: image coding methods based on vector quantization [23], color image quantization [24], and multimedia traffic through MANET by using reinforcement learning techniques [25]. Since ant-based methods found in the literature are not directly applicable to the repurposing service selection problem in the multimedia domain, the focus of this literature review lies on data routing. The ant colony algorithm has been applied to the routing of data packets in different modern networks. Among those, AntNet for data routing [5], AntHocNet for MANET [14] and the biologically inspired routing for MANET [26] are mentionable. However, little is known about the appropriateness for selecting repurposing services among heterogeneous repurposing proxies. As the selection process is somewhat related to routing, the main concept of AntNet, specifically the stochastic selection process is used in selecting repurposing services in our multimedia system.

7 Conclusion

We present a novel biologically inspired repurposing approach for a multimedia system, which makes use of biologically inspired ant colony foraging behavior in multimedia content repurposing. With this proposed system, the main component, the BioReSS algorithm facilitates in selecting repurposing services and optimal service paths based on Quality of Experience by taking both the user's satisfaction (S_i) and the network constraints (network level QoS) into consideration. From the results obtained, bio-inspired solutions to multimedia can potentially fulfill heterogeneous services and the expected user's demands in modern and dynamically changing network environments. We evaluate and compare our proposed system through implementation and simulation. More simulation and implementation are planned to measure the scalability, adaptability, survivability and efficiency of our proposed approach for more multimedia applications over wireless and wired network environments. More subjective measurements of the repurposing system based on the BioReSS algorithm are underway.

References

1. Hossain, M.S., El Saddik, A.: Multimedia content repurposing in wireless environment. In: Furht B. (ed.) Encyclopedia of Wireless and Mobile Communications. CRC Press, Boca Raton (2007)
2. Dijkstra, E.W.: A not on two problems in connection with graphs. Numer. Math. **1**, 269–271 (1959)
3. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms. 2nd edn. MIT Press, Cambridge (2001)
4. Di Caro, G., Ducatelle, F., Gambardella, L.M.: Building blocks from the biology for the design of algorithms for the management of modern dynamic networks. ECRIM (2006)
5. Di Caro, G., Dorigo, M.: AntNet: distributed stigmergetic control for communications networks. J. Artificial Intell. Res. (JAIR) **9**, 317–365 (1998)
6. Dorigo, M., Di Caro, G., Gambardella, L.M.: Ant algorithm for distributed discrete optimization. Artificial Life. **5**(2), 137–172 (1999)
7. Rosenberg, J., et al.: SIP: session initiation protocol. IETF RFC 3261 (2002)
8. Schulzrinne, H., Casner, S., Frederick, R., Jacobson, V.: Real time protocol (RTP): a transport protocol for real-time applications (2003). ftp://ftp.rfc-editor.org/in-notes/rfc3550.txt. Accessed Dec. 2006
9. Jain, R.: Quality of experience. IEEE MultiMedia **11**(1), 95–96 (2004)
10. Hossain, M.S., El Saddik, A.: Proxy-based visual content repurposing using selection algorithm. In: IEEE ICONS 2007, Martinique, French Caribbean (2007)
11. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman, San francisco (1979)
12. Jain, R., Rowe, L.: Acm sigmm retreat report on future directions in multimedia re- search. ACM Trans. Multimedia Comput. Commun. Appl. **1**, 3–13 (2005)
13. JMF:Java Media Framework API (2003) http://java.sun.com/products/java-media/jmf/index.jsp, Acceesed Dec. (2006)
14. Caro, G., Ducatelle, F., Gambardella, L.M.: AntHocNet: an adaptive nature-inspired algorithm for routing in mobile ad hoc networks. Euro Trans. Telecom. (ETT) **16**(5), 443–455 (2005)
15. Di Caro G., Dorigo M.: Two ant colony algorithms for best-effort routing in datagram networks. In: 10th International Conference on Parallel and Distributed Computing and Systems, Las Vegas, Nevada (1998)
16. Information Technology—JPEG 2000 Image Coding System–Part 3: Motion JPEG 2000, ISO/IEC 15 444-3 (2002)
17. Video Coding for Low Bitrate Communication. ITU-T Recommandations H.263 (April 2005)
18. Babaoglu, O. et al.: Design patterns from biology for distributed computing. ACM Trans. Autonom. Adaptive Syst. **1**(1) (2006)
19. BISON Project. http://www.cs.unibo.it/bison/links.shtml, Accessed Dec. 2006
20. Suzuki, J., Suda, T.: A middleware platform for a biologically inspired network architecture supporting autonomous and adaptive applications. IEEE J. Selected Areas Commun. **23**(2), 249–260 (2005)
21. Lee, C., Champrasert, P., Suzuki, J.: iNet: A biologically-inspired adaptation mechanism for autonomic network applications. In: IEEE Upstate New York Workshop on communications and Networking. Rochester, NY (2005)
22. Wedde, H.F., Farooq, M., Zhang, Y.: Beehive: an efficient fault tolerant routing algorithm inspired by honey bee behavior. In: ANTS Workshop, LNCS 3172. Springer, Berlin (2004)
23. Rajpoot, N. et.al.: A novel image coding algorithm using ant colony system vector quantization. In: International Workshop on Systems, Signals and Image Processing (IWSSIP'04)(2004)
24. Hu, X., Wang, T., Li, D.: A New Approach of Color Quantization Based On Ant Colony Clustering Algorithm. In: International Conference on information Technology: Coding and Computing (ITCC-05). Las Vegas, Nevada, USA (2005)
25. Ziane, S., Melouk, A.: A swarm intelligent multi-path routing for multimedia traffic over mobile ad hoc networks. In: 1st ACM international Workshop on Quality of Service and Security in Wireless and Mobile Networks Montreal. Quebec, Canada (2005)
26. Liu, Z., Kwiatkowska, M.Z., Constantinou, C.: A biologically inspired QoS routing algorithm for mobile ad hoc networks: In: Advanced Information Networking and Applications (AINA-2005), Taipei, Taiwan (2005)