AUDIO FINGERPRINTING IN PEER-TO-PEER NETWORKS

Prarthana Shrestha, Ton Kalker
Faculty of Electrical Engineering, Eindhoven University of Technology
P.O. Box 513, 5600 MB Eindhoven, The Netherlands

ABSTRACT

Despite the immense potential of Peer-to-Peer (P2P) networks in facilitating collaborative applications, they have become largely known as a free haven for pirated music swapping. In this paper, we present an approach wherein the collective computational power of the P2P networks is exploited to combat the problem of unauthorized music file sharing. We propose a distributed system based on audio fingerprinting,that makes it possible to recognize the music content present in the network. When the contents are identified, the network can take special measures against the use or sharing of unauthorized music. This proposed system is self-adapting, and robust. The foregoing properties make the system particularly suitable for use in dynamic and heterogeneous environment of P2P networks.

In order to investigate the behavior of the proposed system, a system-level model has been created using the Parallel Object Oriented Specification Language (POOSL). This model was used to investigate an optimal system configuration that maximizes the identification of the content.

1. INTRODUCTION

Peer-to-Peer (P2P) has become an established network for music file storage and sharing and, yet has remained controversial regarding unauthorized music swapping through the network. Apart from legal banning, and use of firewall/ port scanning methods to prevent illegitimate sharing, one of the emerging approaches is the application of Music Information Retrieval (MIR) over P2P networks[1]. The content-based MIR, such as audio-fingerprints and watermarks, can be utilized to identify unauthorized entries in the network; on the basis of which special measures can be taken against their use. In this paper, audio-fingerprinting introduced by [2] is modelled and simulated in P2P network, felicitating distributed computation over the network.

According to the described fingerprinting technique, a stream of audio is converted into a stream of 32-bit sub-fingerprints. A song of 5 minutes produces 25000 sub-fingerprints and thus a moderate database of 10,000 songs gives approximately 250 million sub-fingerprints. In order to identify an unknown audio, a block comprising 256 subsequent sub-fingerprints is used to search for a nearest neighbor match. This means, 250 million searches for the identification of one song! The process becomes far too expensive to execute in a single PC as the database size increases. Therefore, the immense computational power available in P2P network presents an ideal platform to implement distributed audio-fingerprinting.

The fingerprinting technology not only identifies unauthorized files but also provides several benefits to the users. Firstly, it provides efficient browsing; finding an exact match for a search. Secondly, guarantees authenticity on downloads; retrieving exactly as what the name says. And thirdly, the user can use it to organize personal music collection by correct meta-data labelling. However, in this paper we are focused on distributed fingerprinting rather than providing these above mentioned services or integrating with any existing P2P music sharing networks.

The idea of distributed fingerprinting is applied by dividing the large fingerprint database into several segments and distributing them among the peers [3]. While searching for a fingerprint match, they simultaneously check into their database. However, in case of P2P networks, it is to be taken into account that the characteristic of the peers are highly *dynamic*, i.e. unpredictable about joining and leaving the network at any time and *heterogenous*, i.e. the resources possessed by each peer vary in wide range.

In order to coordinate the distributed fingerprinting process with minimum administration, a system is proposed in which the participating peers are divided into hierarchical groups according to their resources. In addition, number of redundant peers are introduced, such that one task is assigned to several peers. This ensures that the result will be available even if some peers disconnect during the job.

The performance of the proposed system was investigated by using a system-level model based on Parallel Object Oriented Specification Language (POOSL). The model was evaluated in terms of latency, query hits and packet loss for various architectures with different peer failure rate. A network with 10 peers including 5 redundant ones, resulted query hits of 94% and latency of 12.3m sec. It was also observed that UDP is the suitable transmission protocol to use in such a system while TCP led to unresolvable deadlocks.

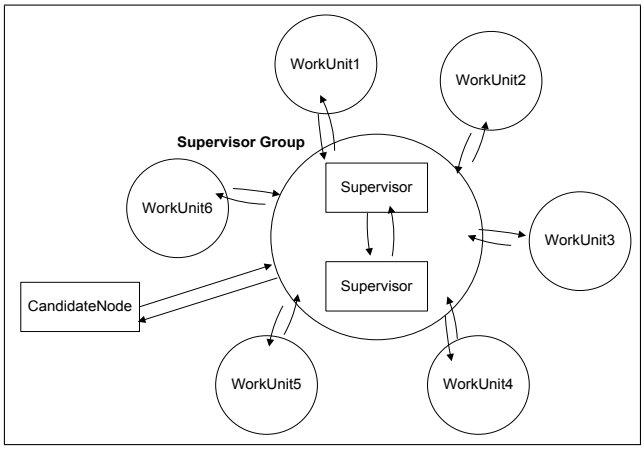

Figure 1. Communication between Supervisor group and Work units.

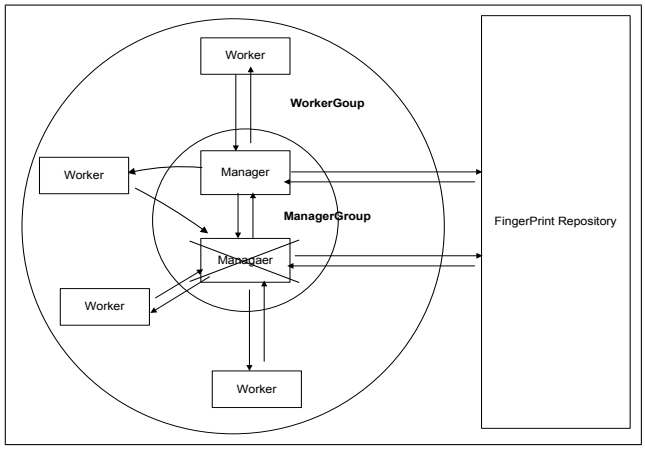

Figure 2. Worker node taking over the role of manager when the latter is disrupted.

2. PEER GROUPS AND FUNCTIONALITIES

Peer groups are fundamental building blocks of the proposed distributed system; holding a specific function, building redundancy, and restricting communication messages only to relevant peers. The system model contains the three following hierarchical peer groups:

The *supervisor* group is a top level group chosen from most resourceful and reliable peers. It co-ordinates the overall activity of the group, such as handling requests for peers joining the network, fingerprint extraction, query formation and submission, and result retrieval. The members of supervisor group communicate with each other in certain intervals and share information that how many peers are working under each of them.

The *manager* group is moderately resourceful and acts as intermediary between supervisor and worker group. Each peer in the manager group possess at least one redundant peer, who owns the same job and data. They are responsible for acquiring and dispatching the fingerprint database fragments and forward query received from supervisor group to worker groups. The group members share information about their status in frequent intervals .

The *worker* group is responsible for performing computations on fingerprint data received from the managers and if a match is found send reply to prescribed address. Each of the workers has one or more redundant peers.The workers are considered as very dynamic and communication among the group members is prohibited.

The peer groups constitute a functional unit called *work unit*, consisting of managers and workers, as an independent entity running the fingerprinting service. In a typical setup, as illustrated in Figure 1, the supervisor sends query to a work unit where a number of managers receive it. The managers forward the query to workers, which are equipped with a segment of the database. The workers search for a match in their database, and a reply is send to the given address. The system is characterized, as described in the following sub-sections.

2.1. Scalability

As more peers join the network, the system should be able to accommodate the increased number of peers without any performance degradation. When the peers join the network first time, they start as workers. As the number workers exceeds certain limit, the manager group invites a resourceful worker to take the manager role. Once the number of managers and workers exceeds the limit, they split into two work units. Similarly, when supervisor gets overloaded, another peer is requested to take the supervisor role; both sharing the work load.

2.2. Adaptability

When some changes occur in the system such as peer failures, the peer groups are expected to adapt themselves without any external intervention and least performance drop. When a worker fails, another redundant worker is expected to complete the job, so no special methods are evoked. When a manager fails, its redundant pair informs the manager group and requests a peer from the worker group, preferably the most resourceful one, to replace the lost manager as described in Figure 2. In case of the supervisor group, at least one of the supervisor is always kept alive, using dedicated peers. If another supervisor is required, an existing supervisor requests the most resourceful peer in the network to take over the role.

2.3. Reliability

Reliability is measured on the basis of correct replies of fingerprint queries within a prescribed time. In order to obtain a reliable system from the dynamic peers, a number of redundant peers are employed who share same responsibility and data.This ensures even if one peer fails, the work is not interrupted.

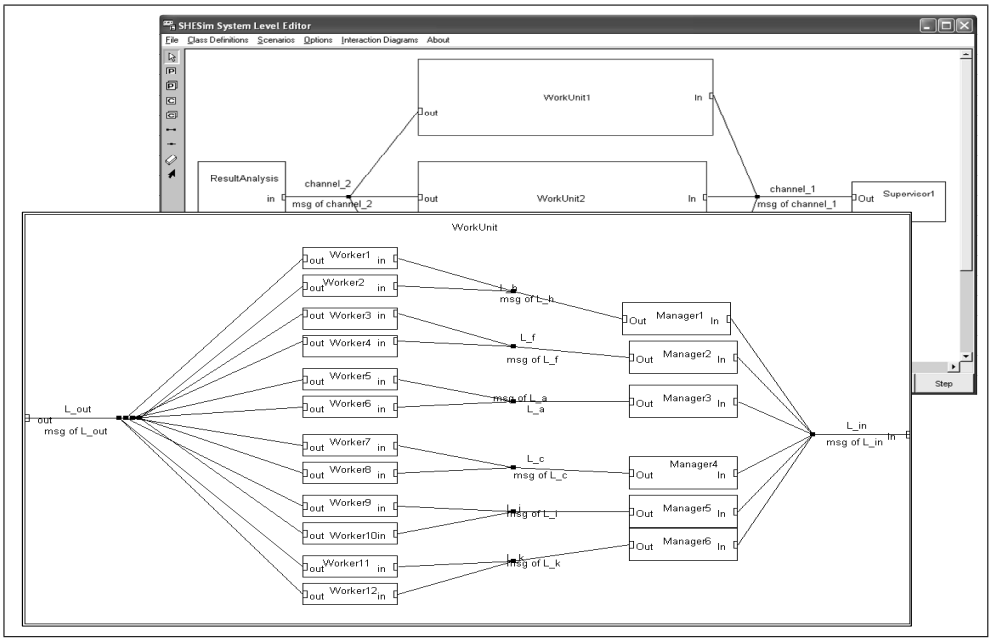

Figure 3. POOSL model of distributed fingerprinting system

3. SYSTEM-LEVEL MODELLING OF DISTRIBUTED FINGERPRINTING SYSTEM

System-level modelling allows an abstract representation of a system, without entailing abstraction from implementation details that are less relevant for analyzing the systems performance. Therefore the proposed system was modelled using Parallel Object-Oriented Language (POOSL) and a graphical simulation tool called SHESim. Once the models were validated in SHESim, they were executed in Rotalumus to achieve fast performance measures. The tools and their detail descriptions are available at [6].

Figure3 depicts a screen-shot of SHESim representing distributed-fingerprinting model. The top window shows the model consisting of two objects, Supervisor and Result Analysis, and a Work Unit. Inspection of a WorkUnit opens up the bottom window, containing Managers and Workers.

This system-level model does not include communication between Manager with database repository nor any internal communications within a peer group and there is also no role switching among peers. The following topics describe the model configurations used for simulation.

Query: Each query is presumed to be an packet size of 1Kbyte, corresponding to 256x32 sub-fingerprint block. The packet framework is illustrated in Figure 4.

PacketID	DestinationID	TimeToLive	EntranceTime	Payload

Figure 4. Query configuration

Transmission channel: P2P applications are built on top of IP (Internet Protocol) and transport protocol, most popularly TCP (Transmission Control Protocol)and UDP. For details please refer to [5] and [4]. In this model a transmission channel is assumed to be 10Mbps wide and UDP and TCP protocols are implemented on it. Based on the experimental data, channel transmission delay calculated to be 2.6msec [5] and packet loss due to UDP as 1% [4].

Peer dynamics: In order to represent peer dynamics, a control variable is set in both Manager and Worker groups that turns off a prescribed number of peers randomly during simulation.

Fingerprint distribution and computation: Each of the Workers is assigned with fingerprint database size of 5000 songs. The model is set with database search time of 4msec on average and computation error of1%, as presented in [2].

4. SIMULATION RESULTS

Performance of the described model was measured using following parameters:

Query Hits: This represents the number of correctly identified songs upon query. Its value depends on the peer failure, transmission errors and accuracy of fingerprint database search algorithm.

Packet Loss: Upon every query, a number of replies are expected depending on the number of redundancies. If the expected packets fail to arrive, due to peer failure or transmission error, they are counted as packet loss.

Latency: Latency is the measure of average time required to get a query reply against transmission and processing delays. It is calculated by using the long run average method available in POOSL, which averages the individual latencies of all the query replies.

The simulations were executed with the confidence level of 95% and accuracy of 99%, based on long-run sample average method available in POOSL. The observed average latency was about 12.3msec and simulation error was $1.5 * 10^{-7}$.

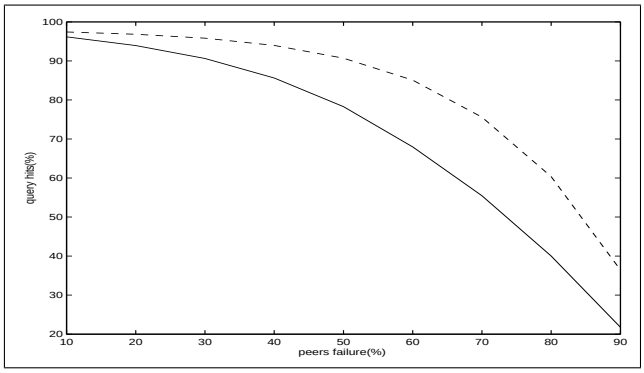

Figure 5. Query hits as a function of peers failure.

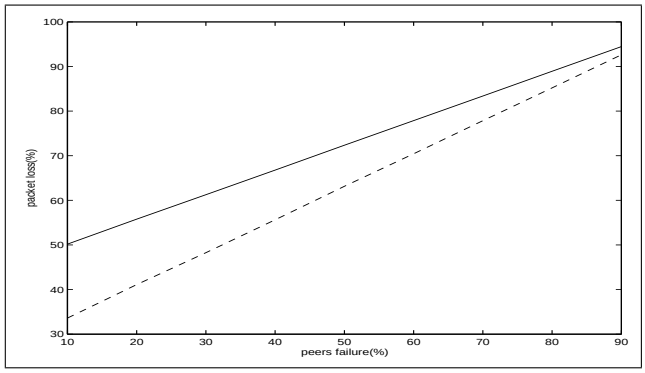

Figure 6. Packet loss as a function of peers failure.

4.1. Results using UDP

In order to investigate the system behavior in different levels of redundancy, the model described in Subsection 3 was executed with different redundancy in same proportion of peer failure; managers 10% and workers 50%. The results are illustrated in Table 1, which were also verified analytically using probability theories.

total manager	total worker	redundant manager	redundant worker	query hits(%)
1	1	0	0	43
2	2	1	0	67
2	4	1	2	87
3	6	2	3	96
4	8	3	4	98

Table 1. Measured query hits for different redundant number of workers and managers under one supervisor

Figure 5 and Figure 6 depict the query hits and packet loss respectively, resulted in varying percentage of failing peers in the system of Figure 3. The simulations were executed in two setups. Firstly, the proportion of workers failure was held 40% and the managers failure was varied, represented by the solid line. Secondly, the manager failure proportion was held 20% and worker failure was varied, represented by dotted line.

4.2. Results using TCP

TCP was implemented using Multicast protocol. Since this involves high connection overhead and requires acknowledgement message upon transmission of each packet, each query ended up in time-out or a deadlock. So no simulation result could be reproduced with this approach.

5. CONCLUSION

In this paper we presented an distributed audio-fingerprinting approach in P2P networks. This enables P2P to become content-aware, which can be used to filter out unauthorized contents from the network. A system was proposed with hierarchical peer architecture and studied its performance in redundancy. The system was simulated and analyzed using a system-level model developed on POOSL. The system behavior can be summarized as follows:
Scalability: the system can be extended to accommodate large number of peers by adding new work units.
Reliability: the system can be made reliable, using adequate redundancy. When 40% of workers and 10% of manager were turned off randomly during simulation, a work unit of 10 peers including 5 redundant ones, resulted 96% query hit, while the packet loss was 50%.
Adaptability: the system can work autonomously and adjust itself in changing circumstances.
Transmission: UDP is an efficient transport protocol for such a system. TCP resulted network deadlocks.
Latency: the average time required to obtain query response is 12.3msec. For a system with more work units, the queries can be processed in parallel.
The future research will focus on evaluating internal communication details and implementation of the model into a system prototype.

6. REFERENCES

[1] Guo J., Tzanetakis G.,Steenkiste P., "Contentbsed retrieval of Music in Scalable P2P Networks", *ICME*, 2003.

[2] Haitsma J., Kalker T., "A Highly Robust Audio Fingerprinting System", *ISMIR*, 2002.

[3] Verbeke J., Nadgir N., Ruetsch G., Sharapov I., "Framework for Peer-to-Peer Distributed Computing in a Heterogeneous, Decentralized Environment", *Sun Microsystems*, July 2002.

[4] Jacklin A., "Using UDP to Increase the Scalability of Peer-to-Peer networks", *M.Sc thesis, University of Sheffield*, May 2003.

[5] Peterson L.L., Davie B.S., *Computer networks: A Systems Approach*, Morgan Kaufmann, 2000.

[6] "POOSL", www.ics.ele.tue.nl/ lvbokhov/poosl/