Virtual Device: Media Service Fitness, Selection and Composition Considering Composition Interactivity and Synchronization

Niall Murray, Brian Lee, A.K. Karunakar, Yuansong Qiao, and Enda Fallon

Software Research Institute, Athlone Institute of Technology, Ireland
{nmurray,ysqiao,akkarunakar}@research.ait.ie,
{blee,efallon}@ait.ie

Abstract. The virtual device enables seamless use of application services residing on different devices in the vicinity of the user. In a pervasive environment, numerous service combinations can be selected to undertake a task. Current works aim to determine the best possible media services for composition by considering user preferences, environment capabilities and similarity between requested and available services. Previously, the authors considered all of above as well as potential local and remote content sources and destination devices. Here this is extended by considering end-to-end service latency to determine service fitness. The end-to-end delay of a service instance is important to consider as it directly affects the interactivity of the system. Services are selected for composition based on our fitness model. We model and simulate this issue and explain the results of our experimentation.

Keywords: Virtual device, Atomic service fitness, service composition, media service selection.

1 Introduction

Today users are surrounded by technology that is heterogeneous, pervasive, and variable [1]. The virtual device combines media services from different devices to support multi-modal communication. Logically it can be viewed as a single device, but each of the services that constitute the virtual device are from different devices. Examples of devices that could provide such services include: small handheld multi-functional devices with limited processing and display capabilities, enhanced single function dedicated devices or powerful multifunctional multimedia systems (e.g. PDA, PCs, HDTV's, network speakers and surround sound systems).

Considering devices in a pervasive environment, multiple instances of the same service types are likely to exist. The suitability (or fitness) of individual services is calculated to distinguish between instances. By selecting and composing the blue-chip services of different devices, the virtual device supports user task satisfaction beyond what a user companion device can offer. A service is defined as an indivisible, self-contained application unit that performs a processing function on multimedia content [2].

L. Atzori, J. Delgado, and D. Giusto (Eds.): MOBIMEDIA 2011, LNICST 79, pp. 281–294, 2012.

The virtual device supports the user task by combining the best service(s) of different devices within the context of the same session. Devices independent of their network connection or platform provide media services (see Fig. 1) to their peers for composition. In general, devices that provide one-to-many services support one service, that is the premium service associated with that device. For example, consider a HDTV that supports video and audio (many TV's currently support many other media services). It's logical to assume that the premium service of this device is its video display considering that people have manually configured surround sound systems in their homes to complement their TV box. Consequently by selecting the premium services from different devices, it is possible to optimally support the user task considering available services. Fig. 1 also includes what we define as content manipulation services, which is outside of scope of this work.

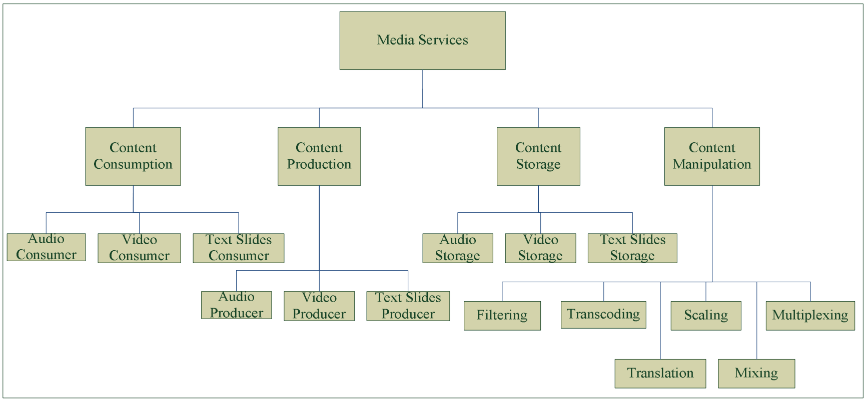

Fig. 1. Multimedia Content Service types [2]

If the goal virtual device is to provide a satisfactory Quality of Service for the user, low delay tolerant scenarios like real-time communication using this type of cluster application must be interactive. It should give the appearance that the selected services of different devices are indeed from a single device. If varying degrees of end-to-end delay exist between the selected services, the virtual device will appear disjointed, with audio/video/other multimedia skews degrading the user experience.

Concurrent services require strict functional and/or timely synchronization [3]. Synchronization ensures that the delivery of media is played out in a timely manner but it can lead to increased latency in a system. The nature of the virtual device, like any other cluster-to-cluster application (3DTI, collaborative workspaces etc.), means that different services available on different devices will have different end-to-end delays. This is because "different sensory streams employ their own protocols and adaptation algorithms in response to the bandwidth dynamics according to their diverse quality-of-service (QoS) requirements" [4]. We consider system end-to-end delay as being effected by network delays [5], network jitter [5] and end system jitters [5].

We model this multi-source, multi-destination service composition challenge using graph theory as per [6][7][8][9].

The goal is to achieve the maximum quality media service composition by considering context. If this selection process is not performed properly, the search will generate non optimized results, causing a low QoS perception from the consumer point of view [10]. Simply choosing the most powerful device or service does not always result in the most efficient and most favourable user experience [2]. It may not be the user's preferred mode of communication. We propose the consideration of estimated end-to-end service delay along with local and remote content consumption and production services, user preferences and device resources. Considering service end-to-end delay results in services being selected that provide greater interactivity in the system. Synchronization schemes like [11] can then be applied to streams with the lowest end-to-end delay. Also as previously proposed [2], considering remote services enables early elimination of unusable services and facilitates reduced media processing costs and delays. Finally, this work provides a method to overcome the dependency [12] on service users' feedback.

The overall contributions of this paper are:

1. Defining the service selection and composition problem to include service end-to-end delay, along with our previous contribution of local and remote services, user preferences and device resources.
2. We model and define the fitness of a service as a function of service end-to-end delay, availability, encodings and potential remote services.
3. We present the results of our experiments displaying the benefits of using the proposed algorithm.

This paper is organized as follows: In section 2 we outline related work. Section 3 defines the problem with the aid of a sample scenario. Section 4 describes how atomic services are modelled in terms of end-to-end delay, availability, compression formats supported and service type. Section 5 presents the algorithms for service composition. Section 6 discusses and explains experimentation and simulation results. Finally in Section 7 this paper is concluded, our contributions are reviewed and our future work explained.

2 Related Work

A broad range of related work involving task based service composition in pervasive environments and MANETS, content delivery and adaptation, the connection of devices in pervasive environments exists.

Similar to this work, Sousa et al. in [1] describe an approach to finding the best match between the user's requirements and the environment's capabilities. Hossain et al. in [13] determine the best possible composition as a function of gain (the extent of which a media service satisfies a user in a particular context) and cost of the service. In [14], Karmouch et al. define service composition in MANETS as a distributed constraint satisfaction problem. Similar to us they use a QoS model based on the work of Pertuttan et al. [15]. Pertuttan's QoS model refers to a degree of match between the user task requirements and the quality of the service composition. This work is based on a per service quality assessment considering the user task. We borrow facets of, and extend Pertuttan's model with end-to-end service delay, device capability and consideration of the remote services. Mukhtar et al. define an approach for task composition considering user preferences and device capabilities [16]. In [8], they use graph theory to define the user task. In [12] Atrey et al. use how regularly a service has been composed with another service, to determine a reputation for that service. In [17] Jiang et al. address service composition based on the prospect of minimum disruption. None of these works consider the end-to-end service delay to select local atomic services.

In relation to selection of the coordinator device, Karmouch et al. in [18] implement a broker based distributed service composition protocol which extends the work of Chakraborty et al. [19]. Basu et al. [7] define graph based approaches to distributed application composition approaches in MANETS.

[20][21][22] propose different approaches for connecting devices in pervasive environments. In [22], Schuster et al. provide a service orientated architecture for virtual device composition utilizing middleware on all devices. In [20], Senthivel et al. construct ad hoc service overlay networks (SON) based on service requests. In [21], Park et al. propose an interoperability framework based on the JXTA protocol. In [23], Ibrahim et al. survey middleware approaches to service composition and define service composition as a four step process: translation, generation, evaluation and execution. In [9], Kalasapur et al. propose a SOA based middleware platform which also incorporates graph theory. In [24], Lee et al. propose an approach based on a virtual device software manager, a middleware manager and hardware adaptation. In [25], Grigoras et al. address MANET formation based on device constraints like bandwidth and battery power.

Much service composition research has focused on media delivery. In [26], Gu et al. propose SpiderNet which provides Statistical QoS assurances for service composition. In [27], Jafarpour et al. strategically place content adaptation nodes in an overlay network to reduce costs in terms of communication and computation. In [28], Qian et al. contribute a multimedia delivery algorithm to calculate the lowest delay path. Xu et al. [29] propose a distributed Storage-assisted Data-driven overlay network to support P2P Video-on-Demand services.

In [30] Nahrstedt et al. introduce and discuss challenges with web services based approaches to multimedia delivery. SPovNet [31] is an overlay based solution that facilitates the spontaneous deployment of distributed network applications and services. In [32], Kim et al. discuss an emerging trend of media orientated service composition with SON's and outline challenges. They also dicuss virtualized resource components as a futuristic solution. In [33], Buford et al. suggest an Internet-scale P2P Overlay to facilitate expanding the capability of a device. Huang et al. in [4][34] propose synchronization solutions for 3DTI where the challenges are similar to those of the virtual device. Boronat et al. in [5] provide a detailed survey on inter stream and group synchronization schemes, whilst in [11] propose a RTP/RTCP based synchronization scheme for cluster-to-cluster applications. In [35] Murray et al. compare SIP and the Advanced Multimedia System as approaches to support future multimedia systems. In [36] Eid et al. incorporate multiple modalities (audio, video, graphics, haptics and scent data) in their Admux framework. In [37], Huang et al. uses prediction to improve interactivity of group synchronization with haptic media. In [38], Ghinea et al. aim to determine perceived synchronization boundaries of olfactory data. In [39] Nunome et al. assess group synchronization schemes for audio and video in wireless ad hoc networks and conclude that a new group synchronization scheme is required. The webinos project [40] aim is to provide a service platform where applications can be used and shared across devices.

To the best of our knowledge, the use of end-to-end delay of potential services has not been a driver to calculate fitness, select and compose local atomic services.

3 Problem Definition

We envisage public and private environments consisting of multiple mobile and stationary nodes that provide content related services. Fig. 2 shows a number of devices providing one-to-many services. All nodes are Internet connected devices where resident services can be invoked by peer nodes. Consider the following scenario defined as real time communication between distributed compositions. Brian is talking with John on his virtual device enabled smart phone and is walking from his office to his car. John is working from home and is using the microphone from his personal computer and surround sound speaker system to converse with Brian. Also in John's study are a PC connected camera and a large display screen. Once Brian sits in his car and places the phone in the holder, his call is automatically transferred from his handheld device to the services available in his car. He is now using the microphone embedded in the sun visor, car radio speakers and can see John on the LCD panel integrated into the car. John can also now see Brian because of the camera built into the steering wheel.

Local services from Brian's perspective are the services resident on devices within his vicinity that he uses to communicate with John (e.g. the microphone embedded in the visor). Remote services from Brian's perspective are the services available to John for communication (e.g. microphone in his personal computer).

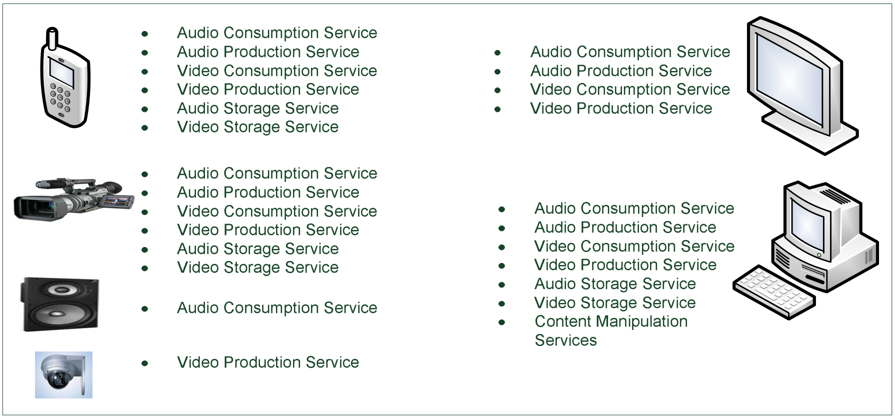

Fig. 2. Shows a number of different devices providing one or many services that could be used to in service composition to create the virtual device

Considering this scenario and Fig. 2, many of the devices in a user's vicinity may provide similar functional services. However differences may exist between these services in terms of capabilities, availabilities, user preferences, usage costs and end-to-end delays. In the scenario outlined, the user may want to use many types of services (e.g. video and audio consumption and production services) in a communication session. Consequently there are multiple constraints in terms of what the user requires, and multiple choices in terms of devices providing different services to solve these problems. Determining the quality of compositions is a variant of a 0-1 Knapsack problem, called multiple dimensional, multiple choice 0-1 Knapsack. Multiple dimensions refer to the multiple constraints and multiple choices refer to choosing one among a set of similar items. In optimizations research this problem has been proven to be NP-complete [41].

Our work aims to achieve the best possible service composition by selecting the fittest instances of the individual service types by considering end-to-end service delay, device resources, user preferences and remote services.

4 Service Composition System

Composing services from distributed devices requires a number of steps. Many of these steps are outside the scope of this document and are outlined in [2]. [2] also provides descriptions of all possible entities required for such a system. It is assumed an Internet scale network exists. Devices provide their content production, storage, consumption and manipulation services for execution by peers. Atomic services provide content processing functionality as per Fig. 1. The end-to-end service delay is the time taken to deliver content from a local content production service to a remote content consumption service or from a remote content production service to a local content consumption service.

4.1 Modelling

This network can be represented by a bipartite digraph, G. This graph is composed of nodes which represent devices that provide one-to-many services. Links in the graph reflect connections between these devices. The disjoint sets in the graph represent the local services and remote services described in section 3. The resources of a device are the quantifiable non-functional characteristics of a device. A scoring of device resources is executed with respect to its peers. Further information on modelling of the network and resources can be obtained in [2].

The services available on a particular node $S = \{S_1, S_2, \ldots S_k\}$ are the content consumption, production, storage and manipulation services as per Fig. 1. The services supported by a device *n* are denoted by $S_i(n)$ where ST(s) represents the

$$S_i(n) = \{ST, E, A, D\} \text{ , where } 0 \leq i \leq k \tag{1}$$

service type, E represents the content compression formats supported and $A(s_i)$ represents the availability status of the service [2]. It is assumed that all nodes in the network can exchange information. $D(s_i)$ represents the end-to-end delay value from a remote site to a local consumption service instance or from a local production service to the remote site. D can be calculated using approaches discussed in [4][11]. End-to-end service delay values are used in algorithm 1 to calculate the fitness of local service instances with respect to their peers. In [2] a specification of a user task is defined. Local and remote content consumption and production services (S_L and S_r respectively), availability, device resources and user preference are considered. Similar to [6][7][8] this specification is detailed using graph theory.

In [2], the contribution was using the existence of remote services and their encoding formats to determine the fitness of local services. The novel aspect of this work is use of the end-to-end delay between potential local services S_L and potential remote services S_r to determine the fitness of local services. The goal is selection of the best possible composition set providing the user with the most interactive user experience.

4.2 Local Atomic Service Fitness Calculation Considering End-to-End Delay

A three step suitability function is used to calculate the fitness of local atomic services of the same type. Per service type ranking tables are generated which reflect the fitness of a service instance with respect to other services instances of the same type. In [2], the work of Pertutten et al [15] was modified and extended to produce local atomic service type ranking tables based on Atomic Service Value ($ASV_t(s_i)$) and Final Service Value ($FSV_t(s_i)$). $ASV_t(s_i)$ considers the devices resources, user inputted weightings, atomic service similarity with requested service, user preferences and availability. As shown in Fig 3, discovery of local services is a prerequisite for this step.

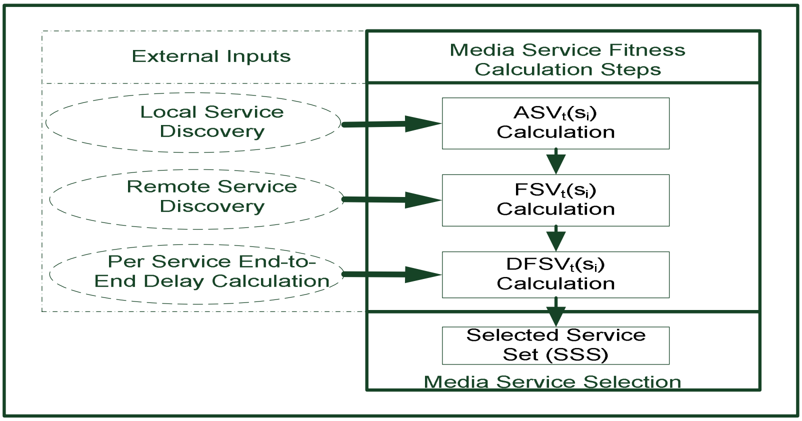

Fig. 3. Service composition system architecture. A number of external entities provide data to the media service composition algorithm. Local service discovery provides all necessary information to calculate and generate tables based on $ASV_t(s_i)$. Remote service discovery provides all data necessary to further calculate and update $ASV_t(s_i)$ ranking tables, producing $FSV_t(s_i)$. Using per service end-to-end delay calculation and $FSV_t(s_i)$ enables the generation of tables based on $DFSV_t(s_i)$. The SSS is calculated based on tables produced after calculation of $DFSV_t(s_i)$.

The $ASV_t(s_i)$ tables are updated by considering existence of potential atomic services of the remote composition resulting in $FSV_t(s_i)$ ranking tables per service type. Again from Fig 3, discovery of remote services is necessary in order to calculate $FSV_t(s_i)$. Detailed information on calculating $ASV_t(s_i)$ and $FSV_t(s_i)$ can be obtained from [2]. This work extends the contribution in [2] to calculate the fitness of local atomic services by considering end-to-end delay. By providing weightings for the services with the lowest end-to-end delays, atomic services that support the most interactive user experience have a greater chance of being selected for composition. As Toshiro et al. have shown in their comparison of the best known group synchronization techniques [15], synchronization, which is necessary in any cluster application like the virtual device increases latency. Considering the delay of a service with respect to its peers results in the generation of updated $FSV_t(s_i)$ ranking tables, namely delay-based final atomic service value ($DFSV_t(s_i)$). This reflects an atomic service's position with respect to other atomic services considering end-to-end delay. As a result of this step, the selected synchronization scheme will be applied to the streams with the lowest end-to-end delays.

4.3 Atomic Service Selection

The final ranking tables comprise atomic services ordered in terms of $DFSV_t(s_i)$. These tables represent the set of all available atomic services for the communication session. The top scoring services in each table are selected for composition. These services make the Selected Service Set (SSS) [2]. Only one service is selected from each of the service tables at any one time. The resultant selected composition reflects the fittest set of services to support the user task based on our service fitness scoring system. The rating for the optimized service composition is a sum of the highest scoring atomic services i.e. sum of the max $DFSV_t(s_i)$ values of each atomic service type.

5 Service Fitness and Composition Algorithm

This section outlines in pseudo code the various steps of the new algorithm introduced as part of this paper. The functionality of this algorithm compliments the algorithms introduced in [2] i.e. calculation of $ASV_t(s_i)$ and $FSV_t(s_i)$. $ASV_t(s_i)$ is calculated by scoring local atomic services in terms of devices resources, user inputted weightings, atomic service similarity with requested service, user preferences and availability. $ASV_t(s_i)$ values are updated firstly by considering existence of remote "partner" services and secondly by determining if commonalities exist between services in terms of the codec's supported. The result is $FSV_t(s_i)$ tables. Pseudo code of algorithms to calculate $ASV_t(s_i)$ and $FSV_t(s_i)$ can be obtained in [2].

Algorithm 1 below compares the end-to-end delay of each of the atomic services within each of the atomic service types. The lowest end-to-end delay values reflect services that will provide the most interactive and real time user experience. Hence, these services are given the highest weighting. $DFSV_t(s_i)$ tables are produced by updating the $FSV_t(s_i)$ tables considering end-to-end delay. For a particular service instance, the delay weighting is multiplied by the respective $FSV_t(s_i)$ value to produce a $DFSV_t(s_i)$ value. The highest scoring atomic services in each of the final set of fitness tables are then selected for composition. It is the highest quality mapping from the user specification to the actual services available on devices. If for any reason, one of the atomic service types becomes unavailable, the next service is selected for instantiation to support execution of the user task.

Algorithm 1: Comparison of local Atomic Services w.r.t end-to-end delay and Optimal Service Composition

```
1.  For each Final service value ($FSV_t(s_i)$) fitness tables needed to
    satisfy a task
2.      For each Atomic Service within each table
3.      Compare its end-to-end delay with other atomic services of
        the same type. //
4.      Determine weighting with respect to peer services.
5.      Update $FSV_t(s_i)$ based on its value with others. Lowest
        end-to-end delay gets highest weighting score to determine
        $DFSV_t(s_i)$ per atomic service.
6.  Sort all rows in table based on $DFSV_t(s_i)$ value.
7.  return All Fitness Tables
8.  For each $DFSV_t(s_i)$ table
9.  SelectTopScoringService()
10. Compose Set of setOfTopScoringServices
```

6 Simulation Results and Analysis

Simulations are performed on a Windows Vista Ultimate OS with 4.00 GB RAM and Intel® Core™ 2 Quad Q6600 @ 2.4GHz. The simulated environment models ten devices that can potentially provide one-to-many services within the vicinity of the user.

For completeness Table 1 below is included which shows a sample generated ranking table for audio consumption type atomic services. All execution points to determine $ASV_t(s_i)$ are displayed. The maximum value for each of the parameters is one, hence the highest possible score of $ASV_t(s_i)$ is also one. Inspecting the values, considering resource capability (R(n)) and similarity of available with requested service $Sim(RS_i, AS_i)$ [15], the PC speakers and surround sound services score as the strongest candidates. All instances are available ($A(s_i)$ column) and hence are given a value of 1. $W_t(s)$ [15], is a user inputted weight to signify how important the user views a service type. The user preferences $U(s_i)$ are the same (value is 0.7) for both the surround sound and PC speakers so after determination of $ASV_t(s_i)$, both services have the same fitness score. Formulas for calculation of $ASV_t(s_i)$ and $FSV_t(s_i)$ are available in [2].

$FSV_t(s_i)$ as shown in table 2 is a the product of columns $ASV_t(s_i)$ and encoding match weighting. Considering encodings supported, the HDTV speakers and surround sound speakers are assigned weightings of 1 as they have common encodings with remote services. The PC speakers do not have a common encoding with remote services, hence it is assigned a weighting of 0.5. At this point the surround sound is considered the best atomic service with a FSV value of 0.7, the HDTV speakers are second strongest with a FSV value of 0.651. These respective values are achieved by multiplying the $ASV_t(s_i)$ and the encoding match weighing to determine the updated service value $FSV_t(s_i)$.

Table 1. Audio Consumption $ASV_t(s_i)$ Fitness table after execution of Algorithm 1

AS Type: Audio Consumer	R(n)	$Sim(RS_i, AS_i)$	$A(s_i)$	$U(s_i)$	$W_t(s)$	$ASV_t(s_i)$
HDTV Speakers	0.93	1.0	1.0	0.7	1.0	0.65
Surround Sound Network connected speakers	1.0	1.0	1.0	0.7	1.0	0.7
PC Speakers	1.0	1.0	1.0	0.7	1.0	0.7
Smart Phone Speakers	0.73	1.0	1.0	0.3	1.0	0.219
Radio Speakers	0.85	1.0	1.0	0.0	1.0	0.0

The final step taken and novel contribution of this work is the consideration of end-to-end delay of an atomic service in determining service fitness using the calculated $FSV_t(s_i)$ and EED weightings as inputs. Different weightings are obtained for each of the audio consumption atomic services based on end-to-end delay. The delay values for each of the atomic services are compared. The lowest delay value between the services gets the weighting value of one.

Table 2. Audio Service fitness calculation: $FSV_t(s_i)$ and $DFSV_t(s_i)$

AS Type: Audio Consumer	$ASV_t(s_i)$	Encoding Match Weighting	$FSV_t(s_i)$	$EED(s_i)$ Weighting	$DFSV_t(s_i)$
HDTV Speakers	0.651	1.0	0.651	0.8	0.5208
Surround Sound connected speakers	0.7	1.0	0.7	0.7	0.49
PC Speakers	0.7	0.5	0.35	0.9	0.315
Smart Phone Speakers	0.219	0.5	0.1095	1.0	0.1095

All other atomic services, ordered in terms of their end-to-end delay, are assigned decremented weightings of 0.1. Hence, the second lowest end-to-end delay is given weighting of 0.9. The third lowest is assigned a weighting of 0.8 etc.

Figure 4a below reflects the ratings of the services as their information is processed through each of the stages of algorithms in this work and in [2]. The services selected for composition are the highest individual scoring atomic services. For explanation purposes, Fig. 4b compares the fitness of the highest scoring SSS with a set comprised of medium scoring atomic services, and two compositions with sets of low scoring atomic services. Comparing the highest SSS with services (s3,s6,s11,s16) and lowest SSS with services (s4,s8,s12,s14) have scores of 2.37 and 0.584 respectively as shown in Fig. 4b.

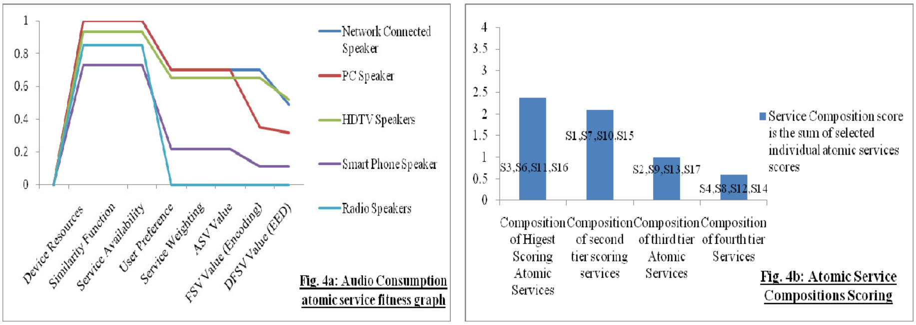

Fig. 4a: Audio Consumption atomic service fitness graph

Fig. 4b: Atomic Service Compositions Scoring

Fig. 4. 4a reflects the change in scores after each of the steps to calculate $ASV_t(s_i)$, $FSV_t(s_i)$ and $DFSV_t(s_i)$. **4b** compares the scores of a set of high fitness services with a set of medium fitness services and a set of low fitness services.

7 Conclusion

This paper has presented an algorithm to calculate service fitness and select services for composition. This novel approach uses the end-to-end delay of local atomic services as input to deciding the local service composition. In addition, it considers device resources, user preferences, remote encodings supported and resource capabilities. The user task is modelled taking into consideration all of the aforementioned factors. A three step suitability process produces ranking tables for required atomic services based on service fitness. The result provides a user with an optimized selection of services, whilst providing an efficient service failure recovery mechanism. The simulation and experimentation show how each of the factors considered; local services, end-to-end delay, remote services, resource capability and user preferences affect the service scoring and how the best possible set of services are selected. In future work, we will investigate synchronization requirements for cluster applications that support multiple correlated media streams, namely audio, video and olfactory data.

Acknowledgments. We would like to recognize the support of Enterprise Ireland through its Applied Research Enhancements program in the financing of this Research. Also Dr. Mark Daly, a lecturer in the AIT School of Engineering for his assistance and advice with system modelling here and in previous work.

References

1. Sousa, J.P., Poladian, V., Garlan, D., Schmerl, B., Shaw, M.: Task-based Adaptation for Ubiquitous Computing. IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews 36, 328–340 (2006)
2. Murray, N., Lee, B., Karanukar, A.K., Qiao, Y., Fallon, E.: Virtual Device: Media Service Fitness, Selection and Composition. In: 4th International ICST Conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications (2011)
3. Nahrstedt, K., Balke, W.-T.: A Taxonomy for Multimedia Service Composition. In: 12th ACM International Conference on Multimedia (2004)
4. Huang, Z., Wu, W., Nahrstedt, K., Rivas, R., Arefin, A.: SyncCast: Synchronized Dissemination in Multi-site Interactive 3D Tele-immersion. In: Multimedia Systems 2011 (2011)
5. Boronat, F., Lloret, J., García, M.: Multimedia group and inter-stream synchronization techniques: A comparative study. Inf. Syst. 34(1), 108–131 (2009)
6. Nahrstedt, K., Yu, B., Liang, J., Cui, Y.: Hourglass multimedia content and service composition framework for smart room environments. Journal on Pervasive and Mobile Computing 1(1), 43–75 (2005)
7. Basu, P., Wang, K., Little, T.D.C.: Dynamic Task Based Anycasting in Mobile Ad Hoc Networks. Journal Mobile Networks and Applications (2003)
8. Mukhtar, H., Belaïd, D., Bernard, G.: A Graph-Based Approach for Ad hoc Task Composition Considering User Preferences and Device Capabilities. In: GLOBECOM Workshops, Current version (2009)
9. Kalasapur, S., Kumar, M., Shirazi, B.A.: Dynamic Service Composition in Pervasive Computing. IEEE Transactions on Parallel and Distributed Systems 18(7) (2007)
10. Dutra, R.G., Martucci Jr., M.: Dynamic Adaptive Middleware Services for Service Selection in Mobile Ad-Hoc Networks. In: Cai, Y., Magedanz, T., Li, M., Xia, J., Giannelli, C. (eds.) Mobilware 2010. LNICST, vol. 48, pp. 189–202. Springer, Heidelberg (2010)
11. Boronat, F., Montagud, M., Guerri, J.C.: Multimedia Group Synchronization for One-way Cluster-to-Cluster Applications. In: 34th IEEE Conference on Local Computer Networks, LCN (2009)
12. Atrey, P.K., Hossain, M.A., El Saddik, A.: A Method for Computing the Reputation of Multimedia Services through Selection and Composition. Journal Mobile Networks and Applications 13(6) (2008)
13. Hossain, M.A., Atrey, P.K., El Saddik, A.: Gain-based Selection of Ambient Media Services in Pervasive Environments. Springer Journal Mobile Networks and Applications (2008)
14. Karmouch, E., Nayak, A.: A Distributed Constraint Satisfaction Problem for Virtual Device Composition in Mobile Ad Hoc Networks. In: Global Telecommunications Conference, GLOBECOM (2009) Current Version (2010)

15. Perttunen, M., Jurmu, M., Riekki, J.: A QoS Model for Task-Based Service Composition. In: Workshop on Managing Ubiquitous Communications and Services (2007)
16. Mukhtar, H., Belaïd, D., Bernard, G.: User Preferences-Based Automatic Device Selection for Multimedia User Tasks in Pervasive Environments. In: 5th Conference on Networking and Services (2009)
17. Jiang, S., Xue, Y., Schmidt, D.: Minimum Disruption Service Composition and Recovery in Mobile Ad hoc Networks. The International Journal of Computer and Telecommunication Networking 53(10) (2009)
18. Karmouch, E., Nayak, A.: A Distributed Protocol for Virtual Device Composition in Mobile Ad Hoc Networks. In: IEEE International Conference on Communications (2009)
19. Chakraborty, D., Joshi, A., Finin, T., Yesha, Y.: Service Composition for Mobile Environments. Mobile Network and Applications 10(4), 435–451 (2005)
20. Senthivel, K., Kalasapur, S., Kumar, M.: PerSON: A Framework for Service Overlay Network in Pervasive Environments. In: Kuo, T.-W., Sha, E., Guo, M., Yang, L.T., Shao, Z. (eds.) EUC 2007. LNCS, vol. 4808, pp. 671–682. Springer, Heidelberg (2007)
21. Park, H., Park, J.-H., Kim, N.: A Framework for Interoperability of Heterogeneous Devices in Ubiquitous Home. In: 2nd Conference on Advances in Future Internet (2010)
22. Schuster, M., Domene, A., Vaidya, R., Arbanowski, S., Kim, S.M., Lee, J.W., Lim, H.: Virtual device Composition. In: 8th International Symposium on Autonomous Decentralized Systems (2007)
23. Ibrahim, N., Le Mouël, F.: A Survey on Service Composition Middleware in Pervasive Environments. International Journal of Computer Science Issues, IJCSI (2009)
24. Lee, J.W., Kim, S.M., Lim, H., Schuster, M., Domene, A.: A Software Architecture for Virtual Device Composition and Its Applications. In: Ubiquitous Computing Systems (2010)
25. Grigoras, D., Riordan, M.: Service Driven Mobile Ad Hoc Networks Formation and Management. In: 4th Symposium on Parallel and Distributed Computing (2005)
26. Gu, X., Nahrstedt, K.: Distributed Multimedia Service Composition with Statistical QoS Assurances. IEEE Transactions on Multimedia 8(1), 141–151 (2006)
27. Jafarpour, H., Hore, B., Mehrotra, S., Venkatasubramanian, N.: CCD: Efficient Customized Content Dissemination in Distributed Publish/Subscribe. In: ACM/IFIP/USENIX International Conference on Middleware (2009)
28. Qian, Z., Guo, M., Zhang, S., Lu, S.: Service-Oriented Multimedia Deliver. In: Pervasive Space. In: IEEE Wireless Communications & Networking Conference (2009)
29. Xu, C., Muntean, G.-M., Fallon, E., Hanley, A.: Distributed Storage-Assisted Data-driven Overlay Network for P2P VoD Services. IEEE Transactions on Broadcasting (2008)
30. Nahrstedt, K., Balke, W.-T.: Towards Building large Scale Multimedia Systems and Applications: Challenges and Status. In: ACM International Workshop on Multimedia Service Composition (2005)
31. The Spontaneous Virtual Network (SpoVNet), `http://www.spovnet.de`
32. Kim, J., Han, S.W., Yi, D.-H., Kim, N., Jay Kuo, C.-C.: Media-Oriented Service Composition with Service Overlay Networks: Challenges, Approaches and Future Trends. Journal of Communications (2010)
33. The Buford, J.F., Yu, H., Lua, E.K.: P2P Networking and Applications. Morgan Kaufmann (2009)
34. Huang, Z., Wu, W., Nahrstedt, K., Arefin, A., Rivas, R.: TSynch: A new Synchronization Framework for Multi-site 3D Tele-immersion. In: 20th International Workshop on Network and Operating Systems Support for Digital Audio and Video, NOSSDAV (2010)

35. Murray, N., Qiao, Y., Lee, B., Karunakar, A.K., Fallon, E.: Design Considerations for Future Multimedia Systems: A Comparison of Approaches Based on SIP and the Advanced Multimedia System. International Journal of Ambient Computing and Intelligence 3(1), 20–32 (2011)
36. Eid, M., Cha, J., El Saddik, A.: Admux: An Adaptive Multiplexer for Haptic-Audio-Visual Data Communication. IEEE Transactions on Instrumentation and Measurement 60(1) (2011)
37. Huang, P., Ishibashi, Y., Fukushima, N., Sugawara, S.: Interactivity Improvement of Group Synchronization Control in Collaborative Haptic Play with Building Blocks. In: 9th Annual workshop on Network and Systems Support for Games (2010)
38. Ghinea, G., Ademoye, O.A.: Perceived Synchronization of Olfactory Multimedia. IEEE Transactions on Systems, MAN, and Cybernetics – Part A: Systems and Humans 40(4), 657–663 (2010)
39. Nunome, T., Tasaka, S.: Inter-Destination Synchronization Quality in a Multicast Mobile Ad Hoc Network. In: 16th IEEE International Symposium on Personal (current version 2006)
40. webinos (Secure Web Operating System Application Delivery Environment), FP7-ICT-2009-5 – Objective 1.2, `http://webinos.org/`
41. Pisinger, D.: An Exact algorithm for Large Multiple Knapsack Problems. European in Journal of Operational Research 114 (1999)