International Scholarly Research Network
ISRN Communications and Networking
Volume 2012, Article ID 340720, 7 pages
doi:10.5402/2012/340720

Research Article

Application of Particle Swarm Optimizer on Load Distribution for Hybrid Network Selection Scheme in Heterogeneous Wireless Networks

Yoke Chek Yee,[1] Su Wei Tan,[2] Heng Siong Lim,[1] and Su Fong Chien[3]

[1] *Faculty of Engineering & Technology, Multimedia University, Jalan Ayer Keroh Lama, 75450 Melaka, Malaysia*
[2] *Faculty of Engineering, Multimedia University, 63000 Cyberjaya, Malaysia*
[3] *MIMOS Berhad, Technology Park Malaysia, 71700 Kuala Lumpur, Malaysia*

Correspondence should be addressed to Yoke Chek Yee, yee.yoke.chek06@mmu.edu.my

Received 23 March 2012; Accepted 19 June 2012

Academic Editors: W.-H. Fang, A. K. Somani, and C.-W. Yi

Mobile terminal with multiradios is getting common nowadays with the presence of heterogeneous wireless networks such as 3G, WiMAX, and WiFi. That Network selection mechanism plays an important role in ensuring mobile terminals are always connected to the most suitable network. In this paper, we introduce and evaluate the performance of load distribution model to facilitate better network selection. We focus on the optimization of network resource utilization using the particle swarm optimizer (PSO) with the objective to distribute the system load according to the various conditions of the heterogeneous networks in order to achieve minimum system cost. Simulation results showed that the proposed approach outperformed the conventional iterative algorithm by a cost improvement of 7.24% for network size of 1000 mobile terminals using 10 particles.

1. Introduction

There has been a drastic and huge development in both mobile technologies such as global system for mobile communications (GSM), general packet radio service (GPRS), and universal mobile telecommunications system (UMTS) which promise high mobility, wide coverage, but low bandwidth rate, as well as on other wireless technologies such as wireless fidelity (WiFi) and worldwide interoperability for microwave access (WiMAX) which offer faster rates at lower cost but suffered from limited mobility and coverage. The different characteristics of these mobile/wireless technologies help compensating for coverage, mobility, bandwidth, and speed, and this helps meeting the requirements due to the increase of user demands in a complementary manner [1]. It is therefore believed that the future network infrastructure will consist of coverage overlapping of heterogeneous networks [2], where multiradios mobile devices could seamlessly and conveniently access to any network in a ubiquitous manner according to the concept of always best connected (ABC) [3].

The challenge to ubiquitous access to any network lies on an efficient and effective mobility management framework which initially focused on enabling seamless vertical handover across heterogeneous networks due to user mobility. Recently, vertical handover is also considered as proactive means to system performance improvement [4, 5]. Realizing a seamless and ubiquitous network access heavily depends on the second phase in vertical handover process called handover decision, which determines and selects one of the most optimal alternative networks to connect to. The selection of network is usually based on parameters such as signal strength, network conditions, battery power, application types, mobile node condition, operator policies, and user preferences [6]. Such selection could be executed by the mobile terminal (MT) in a distributed manner or performed by the network in a centralized manner.

The basic idea of distributed network selection is to enable each MT with capability to receive other network metrics beyond signal information in order to draw conclusion on the best network for subsequent connection. However, such distributed approach failed to consider the load unbalance problem in the entire heterogeneous network [8]. A typical implementation of distributed approach can be found in [9], which is based on session initiation protocol (SIP) following the IEEE media independent handover (MIH) framework, whereby the triggering of handover was mainly based on the signal strength information. Unlike existing distributed approaches which mainly concentrated on network layer performance such as blocking probability and utilization, authors in [10] formulated the heterogeneous networks as a restless bandit system with multimedia application layer performance consideration. Nevertheless, no discussion was made on the overall load balancing across heterogeneous networks.

Load balancing has been considered as one of the key technologies in converged heterogeneous networks [11] as it helps improving resource utilization and scales system capacity and as a result providing better connectivity to users. In [12], load balancing issue for both streaming and elastic applications is studied across both cellular and WiFi networks, whereby streaming application is distributed to cellular network because of its larger coverage and consistent QoS guarantees, while the elastic application is assigned to the WiFi network. In [13], the authors extended a similar study across both WiMAX and WiFi networks, whereby the system is modelled as a G/G/K loss-queue system. However, such study assumed that the capacity of WiFi networks is fixed and had no performance impact on the number of associated MTs. The concept of soft load balancing was also introduced in [13] whereby IP traffic was divided into subflows to be distributed into various different access network based on parameters such as channel conditions, topological environment, multipath, and path loss information overall system performance. There are pros and cons in both distributed and centralized network selection approaches and as such motivated the study of hybrid network selection approaches in recent years [14–16], which in general combine the best schemes/elements from both distributed and centralized approaches into a more effective and efficient network selection framework.

As an effort to support the hybrid network selection approach, we propose a centralized load balancing model to facilitate better network selection. The basis of our model is a system-level cost function which takes into consideration both the network bandwidth and errors to determine the optimal load distribution among heterogeneous networks with minimal system cost. Figure 1 shows an overlay of heterogeneous wireless networks with different cell sizes (e.g., macrocell, microcell, and picocell), similar to the next generation wireless network setup that could be owned by one or multicooperative operators as described in [15] through a central management server. Mobile terminal (MT) is equipped with multiple radios or network interfaces that are capable of attaching to any of these networks. The result of optimal load distribution could then be translated into an adjustment value that will be utilized by the MT to perform network selection as described in the hybrid approach in [7]. Our contribution here is to improve the iterative scheme as proposed in [7] with the evolutionary approach called particle swarm optimizer (PSO) in order to achieve local and possibly global optimal solutions. The choice of particle size also provides extra implementation flexibility on minimal cost versus runtime length.

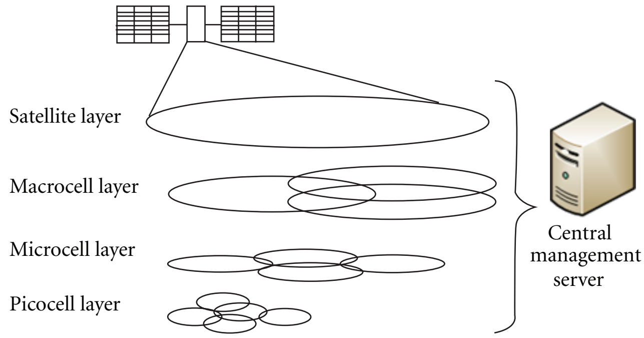

Figure 1: Overlay of heterogeneous wireless networks.

This remainder of the paper is organized as follows. Section 2 describes the derivation of the system cost function. Section 3 presents both the iterative algorithm and the proposed PSO algorithm. The description of the simulation model is provided in Section 4. Section 5 discusses the results, and Section 6 summarizes the paper.

2. Cost Function

Since the proposed PSO approach is a centralized network selection scheme, it is therefore assumed that the cost function at the central load management system for determining optimal load distribution refers to the total cost of all networks (i.e., from satellite, macrocell, microcell to picocell network) on supporting all users' communications [7] as in (1) as follows:

$$\begin{aligned}\mathrm{Cost}(N_1, N_2, \ldots, N_M) &= C_1(N_1) + C_2(N_2) + \cdots + C_M(N_M) \\ &= \sum_{i=1}^{M} C_i(N_i),\end{aligned} \quad (1)$$

where C_i is the cost of ith network, N_i has to pay for supporting MTs which connect to it, and M is the maximum number of networks in the entire system. This means that to support all the users in the network, the ith network has to spend a certain amount of resources such as available bandwidth, processing capability, and computational resource. All these costs are denoted by cost $C_i(N_i)$. In short, C_i is a cost function that relies on the total number of users in network i, represented by (N_i). The larger the network (N_i) is, the larger the cost (C_i) will be. This is because more resources of the network N_i have to be utilized.

The cost function in (1) depends on the total bandwidth offered to all users (B) and the service quality. These two parameters are proportional to the user size in the network. If N_i increases, it implies that the requested bandwidth by the users increases. Since the total bandwidth is limited, the quality of service (QoS) may be degraded if N_i increases.

Hence, to maintain the service quality, the metric of service quality $E_i(\text{err})$ is included in the cost function as well. We define $E_i(\text{err})$ to be the total of all different errors caused by the increase of active user size, network traffic congestion, packet lost, and so forth. Setting all these constraints helps defining a near-realistic heterogeneous network environment whereby heavy traffic shall incur congestion which leads to transmission errors, long service request delays, and maybe link disconnections. However, our model does not include error which is caused by poor channel link or other factors that are not related to the increase of N_i. Hence, C_i in (1) can be written as

$$C_i(N_i) = c_i(b)B_i + c_i(e)E_i(\text{err}), \tag{2}$$

where $c_i(b)$ and $c_i(e)$ are the cost of offering one unit of bandwidth and the cost of correcting each error, respectively. Note that both $c_i(b)$ and $c_i(e)$ are constants that can be selected to make the sum of the values in different units meaningful. We further assume that $B_i = \text{Avg}(B_i)N_i$, where $\text{Avg}(B_i)$ is the average bandwidth requested by each user in network i.

Since $E_i(\text{err})$ is the average number of errors that may occur within a time period, it can be written as

$$E_i(\text{err}) = \sum_{k=1}^{\infty} k p_i(k), \tag{3}$$

where $p_i(k)$ is the probability of having k errors in network i. If q_i is the probability of having only one error during the communication period, then $p_i(k) = q_i^k$, and (3) becomes

$$\begin{aligned} E_i(\text{err}) &= \sum_{k=1}^{\infty} k q_i^k = q_i + 2q_i^2 + \cdots + kq_i^k + \cdots \\ &= \frac{q_i}{(1-q_i)^2}. \end{aligned} \tag{4}$$

Given the probability that a single communication (i.e., of one user) has errors within a time period due to network congestion is a_i, then a single communication with no error within the time period is $1 - a_i$. The probability that N_i communications have no error within one unit time is equal to $(1 - a_i)^{N_i}$. Therefore, the probability that N_i users' communications have errors in a time period can be expresses as

$$q_i = 1 - (1 - a_i)^{N_i}. \tag{5}$$

Thus, (2) can be rewritten as

$$\begin{aligned} C_i(N_i) &= \alpha c_i(b)\text{Avg}(B_i)N_i + \beta c_i(e)\frac{q_i}{(1-q_i)^2} \\ &= \alpha c_i(b)N_i + \beta c_i(e)\frac{q_i}{(1-q_i)^2}. \end{aligned} \tag{6}$$

Note that the $\text{Avg}(B_i)$ can be absorbed into the cost constant $c_i(b)$, and both α and β are the respective weight of the bandwidth and error parameters in the cost function. Both (1) and (6) are the cost functions which will be minimized by the proposed PSO algorithm.

3. Optimization Schemes

As described, it is assumed that the maximum number of MTs in the entire system is fixed for a certain time period, and the number of networks in the system is M. Given $N_1 + N_2 + \cdots + N_M = N$, it means that our defined cost function falls under the nonlinear integer problem since N_i is an integer. Therefore, it is very hard to find the optimal solution for such cost function. We also assume that the cost constants $c_i(b)$ and $c_i(e)$ can be obtained from the central module of the network system, whereby $c_i(b)$ contains the value of $\text{Avg}(B_i)$ that can be obtained from the system according to [17] (e.g., through traffic monitoring/logging over certain period of time). Given the average error probability per unit time of each network a_i, the weight of bandwidth α, the error parameter β, and both cost constants $c_i(b)$ and $c_i(e)$, the proposed network-based selection cost function can be summarized as

$$\begin{aligned} &\text{Given} && \alpha, \beta, c_i(b), c_i(e), a_i \\ &\text{Find} && (N_1, N_2, \ldots, N_M), M \text{ integer variables.} \\ &\text{Minimize} && \text{Cost}(N_1, N_2, \ldots, N_M) = \text{Min}\sum_{i=1}^{M} C_i(N_i) \\ &\text{Subject to} && \sum_{i=1}^{M} N_i = N. \end{aligned} \tag{7}$$

We have evaluated and compared the proposed approach against the iterative algorithm in [7]. Both algorithms are explained in the following subsections.

3.1. Iterative Algorithm. This algorithm works on a simple principle. The network with the highest cost ($C_{\max}$) that is caused by large number of active users should be reduced, whereas the network with the least number of users ($C_{\min}$) should be increased. These two extreme networks will have their δ number of users exchanged at each iterative step. In each iteration, the costs of all networks C_i are recalculated and reordered by cost value. During the next iteration, the next (maybe the same or different) networks with the highest and lowest cost will exchange their δ users. This iteration ends only when the new total cost value is larger than the old one, which was precalculated at the very beginning before the algorithm starts running. See Algorithm 1 for the summary of iterative algorithm.

The results of executing the algorithm will be the optimized users distribution $(N_1, N_2, \ldots, N_M)$ with minimal overall system cost. However, such optimized solution may not guarantee a global optimal solution, and very likely it is just a local minimal solution because it generates only one possible result. To address the limitation of the simple iterative algorithm, we adopt one of the evolutionary algorithms (i.e., PSO) in order to accomplish more optimized results and increase the chances of achieving global optimal solution at the cost of higher complexity as explained in the next few sections. PSO was chosen because it could lead to computational faster convergence than genetic algorithm [18–21].

```
Initialize each N_i with arbitrary value;
Compute Cost_old = Cost_new = Σ_{i=1}^{M} C_i(N_i);
While Cost_old − Cost_new ≥ 0
        Sort all networks by Cost_new value;
        Find C_max and C_min;
        Compute Cost_old = Σ_{i=1}^{M} C_i(N_i);
        N_max = N_max − δ;
        N_min = N_min + δ;
        Compute Cost_new = Σ_{i=1}^{M} C_i(N_i);
End
```

Algorithm 1

3.2. PSO Algorithm. PSO is a well-known evolutionary computation technique developed by Kennedy and Eberhart [22]. PSO is developed according to the social behaviour metaphor. The algorithm is initialized with a population of random candidate solutions conceptually known as particles. Each particle is assigned an arbitrary velocity and will iteratively traverse into the problem space. It is then attracted towards the location of the best fitness achieved so far by the particle itself as well as by location of the best fitness achieved so far by other particles across the entire population. A complete theoretical analysis of the algorithm is presented in [23].

This subsection explains the PSO motion equations. Let $Y_m(t)$ be the position of the mth particle at time t, a vector whose ith component is $y_{m,i}(t)$. Subsequently, $V_m(t)$ is the current velocity of the mth particle at time t, a vector in whose ith component is $v_{m,i}(t)$. Similarly, the best position found by particle m at time t or earlier is represented by $L_m(t)$, which also consists of component $l_{m,i}(t)$. Eventually, $G(t)$ is the best position of all particles at time t or earlier, with component $g_i(t)$. Each particle moves around in space based on the following equations:

$$\begin{aligned} v_{m,i}(t+1) = {} & w v_{m,i}(t) + c_1 \times \text{rand}() \\ & \times \left(l_{m,i}(t) - y_{m,i}(t)\right) + c_2 \times \text{rand}() \\ & \times \left(g_i(t) - y_{m,i}(t)\right) \end{aligned} \quad (8)$$

$$y_{m,i}(t+1) = y_{m,i}(t) + v_{m,i}(t+1), \quad (9)$$

where (8) describes the velocity update for each dimension i of the particle m. The continuous-valued particle swarm algorithm limits $v_{m,i}$ by a value $v_{\max}$, which is a parameter of the system. Subsequently, c_1 and c_2 are two positive constants called acceleration coefficients that affect the maximum size step that each particle can take in a single iteration; rand() is a uniformly distributed random number in the interval [0, 1]. Equation (9) describes the updating of position at ith dimension in particle m. In this study, we use $c_1 = c_2 = 2.0$ as suggested in [24]. The use of the inertia weight w is to speed up the convergence rate in finding the optimal solution. As initially developed, w is often reduced linearly from about 1.2 to 0.4 during a run as in [24]. The correct choice of the inertia weight provides a balance between both global and local exploitation and thus reduces the number of iterations on average in order to find a sufficiently optimal solution. To adapt PSO into the calls distribution optimisation problem, the real value $y_{m,i}$ is rounded up to an integer value, and this integer value is assigned to N_m.

Following is the high-level sequence of the proposed Algorithm 2 which utilizes and calculates the equations described earlier.

Table 1: System parameters.

	$c_i(b)$	$c_i(e)$	a_i
Network$_1$	1.2×10^{-6}	0.12	2.0×10^{-6}
Network$_2$	1.23×10^{-6}	0.113	1.0×10^{-7}
Network$_3$	1.0×10^{-6}	0.12	1.0×10^{-5}
Network$_4$	1.25×10^{-6}	0.11	8.0×10^{-9}

4. Simulation Model

We have conducted a simulation study based on 4 heterogeneous access networks, that is, $M = 4$ as shown in Figure 1. The total number of MTs, $N = 200$, 400, 600, 800, and 1000, are covered by these four networks. It is further assumed that the total number of MTs is fixed during a period of time, with only unforced handover being considered. Both α and β are set to 10000 each, implying that the offering bandwidth and correcting errors are of equal importance to each network. Both cost constants $c_i(b)$ and $c_i(e)$ are obtained from [7] as shown in Table 1.

In this study, we have investigated 3 performance metrics. First is on the effect of applying the PSO algorithm onto achieving minimal system cost as compared to the iterative algorithm in [7], in networks with different number of MTs. We also set a baseline PSO with only two particles as the basic reference, which simulates the general nature of an iterative algorithm, against the proposed flexibility of PSO configurations with more than two particles. Second is on seeking for further understanding on the effect of particle size onto achieving minimal system cost. Last is on investigating the use of PSO with different particle size for network with large number of users.

5. Results and Discussion

Table 2 shows the optimal calls distribution for N from 200 to 1000, with their respective minimum costs, and the actual calls distribution size across the 4 networks using 10 particles. In general, the minimal cost increases as the network size grows as shown in Figure 2. The calls distribution is not evenly spread across all 4 networks with the highest loads always on network 4. This is because it has the smallest a_i, that is the network with the lowest probability of having errors due to network congestion. The same explanation goes to network 3 for having the lowest or zero loads. Both $c_i(b)$ and $c_i(e)$ did not bring any significant effect to the overall distribution because their values are not much different across the 4 networks. Hence, it is concluded that

Create and initiate P-particles of M-dimension and their associated objective functions (5) and (6),
where P is the number of particles, and M is the number of networks
Repeat:
  For each particle $m \in [1, 2, \dots, P]$, where P is the number of particles
    For each dimension $i \in [1, 2, \dots, M]$, where M is the number of networks
      Calculate (8) and (9)
    end for;
    Update the fitness value, which is the system cost value
    Update the best position found so far (Global and Local) using the fitness value;
  end for particles;
until a satisfactory solution is reached or computational limit are exceeded.

Algorithm 2

Table 2: Minimum cost and calls distribution for different network sizes.

Network size	Min. cost	$N1$	$N2$	$N3$	$N4$
200	2.54	19	7	0	174
400	5.42	4	34	43	319
600	7.66	87	143	0	370
800	10.05	26	73	0	701
1000	13.06	218	83	16	683
1000 [7]	14.08	240	319	121	320

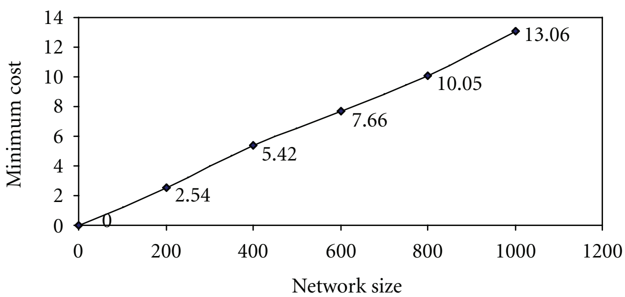

Figure 2: Minimum cost versus network size.

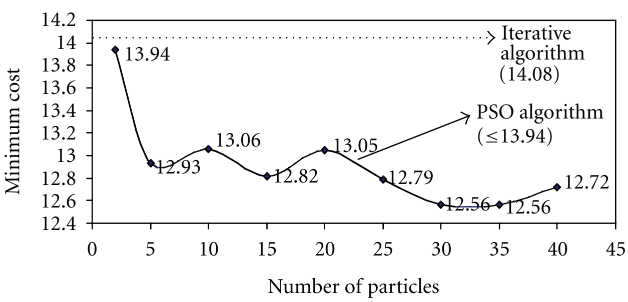

Figure 3: Minimum cost over the number of particles.

a_i carries the highest weight in determining the optimal calls distribution.

Detailed calls distribution for each of the 10 particles is shown in Table 3 sorted in ascending order based on the minimal cost achieved.

As compared to the results from [7], which achieved the minimal cost of 14.08 with the distributions of 240, 319, 121, and 320 as indicated in the last row of Table 2, it clearly shows that the PSO algorithm with a minimal cost of 13.06 has outperformed the iterative algorithm. The limitation of the iterative algorithm lies very much on the initial calls distribution across the networks (which determines the initial cost), and also on a single walkthrough in the problem space guided by the initial cost calculated. Such approach would most likely achieve only local optimal solution. Unlike the iterative algorithm, PSO employs a variable number of particles to seek for optimal solution, which helps increasing the chance of achieving global optimal solution, that is, a cost improvement of 7.24% for network size of 1000 MTs.

Table 4 shows the optimal calls distribution of utilizing different number of particles, from 2, 5, 10, 15, 20 to 40, with their respective minimum costs, and the actual calls distribution size across the 4 networks for 1000 MTs. In general, the minimal cost decreases as the particle size grows, but with some interesting exceptions as shown in Figure 3. For example, utilizing 10 particles seems to incur slightly higher cost (close to 1%) than 5 particles, that is, failed to reduce the overall system cost further. The same observation goes to the case with 20 and 15 particles. However, based on the results in Table 2, PSO algorithm should still be a favourable approach here for achieving better optimization with very minor glitches on particle size configurations.

6. Summary and Future Works

This paper has formulated the cost function for calls distribution in heterogeneous networks based on two parameters,

Table 3: Minimum cost and calls distribution for 10 particles.

Particle ID	Min. cost	$N1$	$N2$	$N3$	$N4$
1	13.06	218	83	16	683
2	14.92	218	212	212	358
3	14.94	217	214	214	355
4	14.95	218	215	215	352
5	14.95	215	220	216	349
6	14.95	215	216	216	353
7	14.96	219	213	216	352
8	15.11	230	229	229	312
9	15.23	240	240	240	280
10	15.29	244	250	246	260

Table 4: Minimum cost and calls distribution for different particle sizes.

Particle size	Min. cost	$N1$	$N2$	$N3$	$N4$
2	13.94	248	173	103	476
5	12.93	75	178	31	716
10	13.06	218	83	16	683
15	12.82	158	24	2	816
20	13.05	3	217	59	721
25	12.79	72	24	16	888
30	12.56	9	226	6	759
35	12.56	3	186	7	804
40	12.72	53	80	13	854

namely, the offering bandwidth and service quality (i.e., probability of having communication errors) and presented an approach using PSO to seek for minimal system cost. The proposed approach outperformed the iterative algorithm in the literature by a cost improvement of 7.24% for network size of 1000 MTs using 10 particles. Our proposed approach has higher chance of achieving global optimal solution as compared to the iterative algorithm given the option to utilize a variable number of particles in the problem space.

One possible future work is to consider more parameters besides bandwidth and service quality, and this includes application types, battery power, operator policies, and perhaps user preferences. It is also interesting to study the load balancing issues with different theories and models such as evolutionary games for performance comparison.

References

[1] J. S. Kim, E. Serpedin, D. R. Shin, and K. Qaraqe, "Handoff triggering and network selection algorithms for load-balancing handoff in CDMA-WLAN integrated networks," *Eurasip Journal on Wireless Communications and Networking*, vol. 2008, Article ID 136939, 2008.

[2] M. M. Alkhawlani and K. A. Alsalem, "Radio network selection for tight-coupled wireless networks," in *Proceedings of the 7th International Conference on Informatics and Systems (INFOS '10)*, Cairo, Egypt, March 2010.

[3] N. Passas, S. Paskalis, A. Kaloxylos et al., "Enabling technologies for the always best connected concept," *Wireless Communications and Mobile Computing*, vol. 6, no. 4, pp. 523–540, 2006.

[4] X. Liu, V. O. K. Li, and P. Zhang, "Joint radio resource management through vertical handoffs in 4G networks," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM '06)*, December 2006.

[5] A. E. M. Taha, H. S. Hassanein, and H. T. Mouftah, "Vertical handoffs as a radio resource management tool," *Computer Communications*, vol. 31, no. 5, pp. 950–961, 2008.

[6] Y. C. Yee, K. N. Choong, A. L. Y. Low, and S. W. Tan, "SIP-based proactive and adaptive mobility management framework for heterogeneous networks," *Journal of Network and Computer Applications*, vol. 31, no. 4, pp. 771–792, 2008.

[7] X. Jiang, *Mobility management in next generation all-IP based wireless systems [Ph.D. thesis]*, Georgia Institute of Technology, Atlanta, Ga, USA, 2004.

[8] H. Jia, Z. Zhang, P. Cheng, H. H. Chen, and S. Li, "Study on network selection for next-generation heterogeneous wireless networks," in *Proceedings of IEEE 17th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '06)*, September 2006.

[9] K. N. Choong, V. S. Kesavan, S. L. Ng, F. de Carvalho, A. L. Y. Low, and C. Maciocco, "SIP-based IEEE802.21 media independent handover—A BT Intel collaboration," *BT Technology Journal*, vol. 25, no. 2, pp. 219–230, 2007.

[10] P. Si, F. R. Yu, H. Ji, and V. C. M. Leung, "Optimal network selection in heterogeneous wireless multimedia networks," in *Proceedings of IEEE International Conference on Communications (ICC '09)*, Dresden, Germany, June 2009.

[11] B. Li, W. Shi, and Y. Zhao, "A load balancing algorithm based on dividing IP flow for high-speed traffic over heterogeneous

wireless networks," in *Proceedings of the 3rd International Congress on Image and Signal Processing (CISP '10)*, pp. 4294–4298, Yantai, China, October 2010.

[12] W. Song and W. Zhuang, "Multi-service load sharing for resource management in the cellular/WLAN integrated network," *IEEE Transactions on Wireless Communications*, vol. 8, no. 2, pp. 725–735, 2009.

[13] J. Xu, Y. Jiang, A. Perkis, and E. Khloussy, "Multi-service load balancing in a heterogeneous network with vertical handover," in *Proceedings of the 1st European Teletraffic Seminar*, Poznan, Poland, 2011.

[14] W. Song, W. Zhuang, and Y. Cheng, "Load balancing for cellular/WLAN integrated networks," *IEEE Network*, vol. 21, no. 1, pp. 27–33, 2007.

[15] M. M. Alkhawlani and A. A. Hussein, "Intelligent radio network selection for next generation networks," in *Proceedings of the 7th International Conference on Informatics and Systems (INFOS '10)*, Cairo, Egypt, March 2010.

[16] D. Niyato and E. Hossain, "Dynamics of network selection in heterogeneous wireless networks: an evolutionary game approach," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 4, pp. 2008–2017, 2009.

[17] H. J. Wang, R. H. Katz, and J. Giese, "Policy-enabled handoffs across heterogeneous wireless networks," in *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '99)*, pp. 51–61, February 1999.

[18] X. Wu, S. Lei, J. Yang, H. Xu, J. Cho, and S. Lee, "Swarm based sensor deployment optimization in Ad Hoc sensor networks," in *Proceedings of the 2nd International Conference on Embedded Software and Systems*, pp. 533–541, Xi'an, China, 2005.

[19] S. Pandey, L. Wu, S. M. Guru, and R. Buyya, "A particle swarm optimization-based heuristic for scheduling workflow applications in cloud computing environments," in *Proceedings of the 24th IEEE International Conference on Advanced Information Networking and Applications (AINA '10)*, pp. 400–407, April 2010.

[20] M. Khodier and M. Al-Aqeel, "Linear and circular array optimization: a study using particle swarm intelligence," *Progress In Electromagnetics Research B*, no. 15, pp. 347–373, 2009.

[21] S. Deb, C. J. Singh, N. B. Singh, A. K De, and S. K. Sarkar, "Parameter optimization of quantum well nanostructure: a PSO and GA based comparitive study," *International Journal of Latest Trends in Computing*, vol. 35, no. 1, pp. 35–40, 2010.

[22] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, vol. 1, pp. 1942–1948, December 1995.

[23] M. Clerc and J. Kennedy, "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 58–73, 2002.

[24] C. F. Teo, Y. C. Foo, S. F. Chien, A. L. Y. Low, A. H. You, and G. Castañón, "Wavelength converters placement in all optical networks using particle swarm optimization," *Photonic Network Communications*, vol. 10, no. 1, pp. 23–37, 2005.