A Bubble Oscillation Algorithm for Distributed Geographic Load Balancing in Mobile Networks

Lin Du, John Bigham, and Laurie Cuthbert
Department of Electronic Engineering
Queen Mary, University of London
London E1 4NS, United Kingdom
Email: {lin.du, john.bigham, laurie.cuthbert}@elec.qmul.ac.uk

Abstract—This paper investigates a new load balancing scheme for mobile networks that changes cellular coverage according to the geographic traffic distribution in real time. The performance of the whole cellular network is improved by contracting the antenna pattern around a traffic "hot spot" and expanding adjacent cells coverage to fill in the coverage loss. This is essentially a constrained multidimensional optimization problem. A novel bubble oscillation algorithm is proposed to address this problem. Any un-served traffic in the network is absorbed by the geographic load balancing in a similar way that a vacuum between bubbles is filled by bubble oscillations. Simulations have been performed to evaluate the system performance for different traffic scenarios, and the results are presented. Some discussions on this algorithm are also presented. The bubble oscillation algorithm described has the potential of being used in other similar multidimensional resource allocation problems.

Keywords—System design, Simulations, Mathematical programming/optimization.

I. INTRODUCTION

Mobile cellular networks are by far the most common of all public wireless communication systems. Previous research on mobile cellular networks has led to many schemes to increase the system capacity. Balancing the traffic load [1] and use of smart antennas [2] are two of the most important ones. Traffic load balancing in mobile cellular network has been well-studied since the first generation of mobile communication systems. Many methods have been proposed to address this problem, such as cell splitting [1], channel borrowing [1], [3], channel sharing [4], dynamical channel allocation [5]–[7], new soft handover schemes [8], [9], etc. The applications of smart antennas in cellular networks are also widely investigated, such as [10]–[12]. However, most work related to traffic load balancing only focuses on different radio channel allocation schemes, and most work on smart antennas only considers the radio propagation channels within one cell. These severely limit their efficiency.

Geographic load balancing is recognized as a new approach for traffic load balancing which provides dynamic load redistribution in real time according to the current geographic traffic conditions. It can be used to improve the performance for any distributed systems containing non-uniformly distributed traffic, especially for resolving the traffic hot spots. Studies on dynamic sectorization [13], use of tilted antennas [14], and dynamic cell-size control (cell breathing) [15], [16] have shown that the system performance can be improved by balancing non-uniformly distributed traffic. Our previous work using cooperative negotiation techniques and semi-smart antenna to provide dynamic geographic load balancing for mobile cellular networks has been shown to be effective [17], and geographic load balancing optimization using evolutionary computations have provided a performance benchmark [18]. The formation of cells is based upon call traffic needs. Capacity in a heavily loaded cell can be increased by contracting the antenna pattern around the source of peak traffic and expanding adjacent antenna pattern to fill in the coverage loss as illustrated in Fig. 1.

Fig. 1. Geographic load balancing in mobile cellular networks.
to be performed cooperatively, as the local base stations have very limited capability of resolving traffic hot spots independently. The best-fit antenna pattern from the available antenna resource is then synthesized using a pattern synthesis method, which takes into account the physical constraints of the antenna, and finally make the physical coverage change accordingly. In this paper, we will focus on how to decide the optimum local coverage pattern in the context of the whole network.

A novel bubble oscillation algorithm is introduced as a distributed optimization technique to address this problem. The local coverage scheme is treated as an air bubble, the local traffic load is treated as the air within the bubble, and the un-served traffic is treated as the vacuum between adjacent bubbles. Thus, the process of load balancing is performed by emulating the process of bubble oscillation, and the attempts of re-allocation of un-served traffic are fulfilled by the oscillations caused by the pressure difference between adjacent cells or attraction forces from temporary vacuums.

This paper is organized as follows. In Section II, the geographic load balancing in mobile cellular networks is first formalized as a multi-dimensional resource allocation problem.

The bubble oscillation algorithm is described in Section III to address the geographic load balancing problem. The bubble oscillation algorithm described here has the potential of being used in other similar multi-dimensional resource allocation problems.

System level simulations have been performed to evaluate the system performance for different traffic scenarios. The simulations are described and the results are presented in Section IV.

In Section V, some techniques that can improve the performance of bubble oscillation algorithm are discussed. They have provided some basic guidelines for reducing the oscillation time and speeding up the convergence.

II. GEOGRAPHIC LOAD BALANCING IN MOBILE CELLULAR NETWORKS

Suppose that a mobile network contains \( m \) base stations and \( n \) traffic units (users that are currently talking or transmitting data). Each base station has constraints, such as its maximum capacity, the maximum distance it can transmit a signal, the physical limit of its transmitting power. The distance from a traffic unit to its subscribed base station also needs to be minimized, so as to save energy and reduce the interference to others. The geographic load balancing is a process that maximizes the system capacity by equalizing the \( n \) traffic units over \( m \) base stations, and satisfying all these constraints.

We can then formalize this as a multidimensional optimization problem with some constraints. The objective is to increase the system capacity while keeping the transmission power at the minimum. The constraints are listed as follows.

1) The total served demand at any base station should be less than its capacity;
2) The total transmission power at any base station should be less than its maximum transmitting power;
3) At most \( M \) base station can serve one traffic unit (represents soft handover), where \( M \) is defined by different network topologies;
4) The distance from a base station to any served traffic unit should be shorter than the maximum distance the base station can transmit signals.

More explicitly, this can be formulated as (1) and (2).

\[
\begin{align*}
1. \text{Maximize } & \sum_{j=1}^{n} c_j \cdot x_{ij}, \\
2. \text{Minimize } & \sum_{j=1}^{n} a_{ij} \cdot x_{ij},
\end{align*}
\]

subject to

\[
\sum_{j=1}^{n} c_j \cdot x_{ij} \leq b_i, \forall i = 1, \ldots, m, \\
\sum_{j=1}^{n} a_{ij} \cdot x_{ij} \leq p_i, \forall i = 1, \ldots, m, \\
\sum_{i=1}^{m} x_{ij} \leq M, \forall j = 1, \ldots, n, \\
d_{i,j} \cdot x_{ij} \leq d_{i,\text{max}}, \forall x_{ij} = 1, \\
x_{ij} \in \{0, 1\}, 1 \leq j \leq n, 1 \leq i \leq m.
\]

where \( n \) is the number of traffic units, and \( m \) is the number of base stations. \( m \) is also the dimensions of the constraints. \( c_j \) represents the demand of the \( j \)-th traffic unit, \( x_{ij} \) is a binary variable that indicates if the \( j \)-th traffic unit has been served by the \( i \)-th base station (\( x_{ij} = 1 \)) or not (\( x_{ij} = 0 \)), \( b_i \) represents the \( i \)-th base station’s capacity, \( a_{ij} \) is the transmitting power required at the \( i \)-th base station for serving the \( j \)-th traffic unit, \( p_i \) represents the \( i \)-th base station’s maximum transmitting power, \( M \) is the maximum number of base stations that can serve a traffic unit, \( d_{i,j} \) is the distance from the \( j \)-th traffic unit to the \( i \)-th base station, and \( d_{i,\text{max}} \) is the maximum distance allowed by the \( i \)-th base station.

This problem is similar to multidimensional knapsack problem (MKP) [20], which is a NP-hard problem and does not have a universal solution yet. The dual objective along with the complicated constraints make our problem even harder than the standard MKP. These have urged us to explore new methods, by which the special characteristics of geographic load balancing can be better utilized to obtain more accurate results with shorter computation time.

From our previous work of using evolutionary computation [18] and cooperative negotiation techniques [17], we have learned that both local optimizations and inter-cell communications are very important to solve this constrained multi-dimensional optimization problem. Inspired by the analogy between bubbles and cells, we propose a bubble oscillation algorithm to address the geographic load balancing problems, which will be described with more details in the next section.

III. BUBBLE OSCILLATION ALGORITHM

Since our solution to the distributed geographic load balancing problem for mobile cellular networks is based on the
emulation of bubble oscillations, a brief review of the physical behavior of some interconnected bubbles is given first. The similarities between them are described next, and a novel bubble oscillation algorithm is derived from the analogy to address the given problem. An example is also given to explain this algorithm.

A. Behavior of Interconnected Bubbles

Suppose there are seven air bubbles (B0 to B6) on a plane, interconnected with each other, as shown in Fig. 2. Each bubble contains certain amount of air, and its shape is distorted by the pressure from adjacent bubbles.

Let us assume that the outer contour of the seven bubbles is fixed (e.g. within a container) and they have already reached a balance before our analysis. If a small amount of air in bubble B1 is drawn out very quickly, the air pressure within B1 is reduced suddenly. As the shapes of other bubbles have not changed yet, some vacuum will appear around bubble B1. For simplicity reason, we only use a vacuum V0 to explain the re-balancing process. It is shown as the shadow area at the boundary of bubbles B0 and B1 in Fig. 2.

According to physical rules, the vacuum V0 first generates two attraction forces F0 and F1 to B0 and B1 respectively, and bubbles B0 and B1 change their surface to fill the vacuum. However, this reduces the air pressure of both B0 and B1, which breaks the pressure balance at the boundary of other bubbles. The pressure difference generates attraction forces F2, F3, ..., and F6 for B2, B3, ..., and B6 respectively, and causes them change their size and shape to balance the air pressure between adjacent bubbles. Unfortunately they cannot reach a new balance just from one turn of adjustment, as the attraction forces are generated according to the difference of their local air pressure at a specific time. Once any bubble’s size and shape changes, its air pressure and the attraction or repulsion forces for adjacent bubbles will change too. If this happens in an ideal environment, the bubble oscillation will last for ever. However, in the existence of damping in the real life, the amplitude of oscillation becomes smaller and smaller, and finally reach a new balance after some iterations.

B. Bubble Oscillation Algorithm

There are several similarities between bubble oscillation and geographic load balancing in mobile networks, which makes it attractive to model the cellular network as inter-connected bubbles.

1) The bubble surface is quite similar to the cell contour,
2) The air within each bubble can be analogous to the traffic served by each cell,
3) The volume of each bubble is similar to the capacity of each base station,
4) The temporary vacuum can be related to any un-served traffic, and
5) The process of bubble oscillations is similar to the adjustments of geographic load balancing for mobile cellular networks.

Based on these similarities and the fact that naturally bubble oscillation can reach a new balance automatically, we propose a bubble oscillation algorithm for geographic load balancing in mobile networks. According to physics theory, unbalanced bubbles can always reach a new balance through oscillations at the existence of any damping. Therefore, the convergence of this bubble oscillation algorithm can be guaranteed as long as there is some mechanism that can reduce the oscillation amplitude for each iteration. This damping mechanism is realized by a simple scheduling method, which will be described later.

As the cellular coverage is no longer fixed, it is inadequate to simply model the coverage area by hexagonal cells. We define the cell frontier as the maximum outreach of its base station can transmit, as shown in Fig. 3. Any traffic outside the
frontier is of no direct interest to the base station, as it cannot service any of it. Therefore, the fourth constraint described in Section II is satisfied automatically. The hexagonal region in the figure represents the default, baseline pattern of coverage, the small ring represents the region that can only be covered by its own base station, and the shadowed part in the middle represents the forbidden zone (no mobiles in it). Local polar coordinates are used for the advantages of modelling antenna radiation patterns, and global cartesian coordinates are used for the location of users and communication between adjacent cells.

As shown in Fig. 3, we first define an initial repulsion force \( \vec{R}_{ij}^{0} \) generating from the \( i \)-th base station to the \( j \)-th traffic unit within its frontier. It is expressed as (3) in the local polar coordinates.

\[
\vec{R}_{ij}^{0} = (d_{i,\text{max}} - d_{ij}, \theta_{ij}),
\]

where \( d_{ij} \) and \( \theta_{ij} \) is the distance and angle from the \( i \)-th base station to the \( j \)-th traffic unit.

The radial repulsion force represents how interested the base station is in having the specific traffic unit assigned to it. As the repulsion force decreases along the distance, the traffic units closer to base station have higher priority to be served. This has included our second optimization objective, which minimizes the base station transmitting power by favoring the closer traffic units.

The cell contour is drawn in such a way that the magnitude of the repulsion forces of traffic units on the contour are equal to a threshold value of \( |\vec{R}_{iT}| \), the amount of traffic inside is equal to or less than to its capacity, and the total transmitting power is equal to or less than its maximum power. The first and second constraints mentioned in Section II are enforced during the process of forming cell contours.

Calculating the threshold \( |\vec{R}_{iT}| \) is a local optimization problem. The local list of traffic units is sorted by their repulsion force into descending order, then we check if all the constraints are satisfied from the first traffic units to the last one in the sorted list. The threshold is decided by the repulsion force of the last one that can still satisfies all the constraints. This local optimization process is described in the flowchart in Fig. 4. This is not the most efficient way of doing local searching, but it is the simplest one. Better searching algorithms can be used to improve the computation performance, however it is not really necessary as the length of those local lists are reasonably short. With the initial repulsion force values, all the cell contours are circular. However, they will be distorted according to traffic conditions after the adjustments start, as described next.

The formation of cells is related to bubbles with the similarities between repulsion force and air pressure, as both forces have the tendency to expand the size. They also share the similar elasticity features, which can restore the default shape automatically and avoid producing any unrealizable patterns for the available antenna resource. Unlike the bubbles, the cell contours can overlap with each other, as shown in Fig. 3. The number of overlapping layers may vary for different cellular network topologies, but this does not invalidate the proposed algorithm. Thus any traffic unit could be served by several base stations at the same time (soft-handover), which has satisfied the third constraint described in Section II.

The main objective, to maximize the system capacity, is realized by a process of reducing the number of un-served traffic units. This is similar to the process of eliminating vacuums with bubble oscillations as described before. After the initial local optimizations, it is possible that some traffic units will not be served by any base stations. The attraction forces are generated from them in order to adjust the initial repulsion forces of the traffic units nearby. The local opti-
mizations are re-performed with new repulsion forces. This process is iterated until the results converges (i.e. the number of un-served traffic keeps unchanged for some iterations or becomes zero) or the predefined maximum iteration number has reached. This is shown as the main loop in Fig. 4.

The cell contours are distorted by the variable attraction forces from the un-served traffic units iteratively, and finally obtain the optimum solution for current traffic condition. Fig. 5 shows an example of seven cells where an un-served traffic unit is caused by traffic hotspots at the boundary of BS0 and BS1. The details of bubble oscillation algorithm resolving the traffic hotspots problem are described next. The calculation of attraction forces from un-served traffic units and new repulsion forces are also explained.

![Fig. 5. Seven cells with an un-served traffic unit.](image)

The attraction force from the k-th un-served traffic unit V_k to the j-th traffic unit T_j at the i-th base station BS_i during the λ-th iteration is defined as \( \vec{A}_{kj}^{\lambda} \) in the local polar coordinates,

\[
\vec{A}_{kj}^{\lambda} = \begin{cases} (w_0, \theta_{ij} - \theta_{ik}), & k \neq j \\ (w_0, \theta_{ij}), & k = j \end{cases},
\]

where \( \theta_{ik} \) and \( \theta_{ij} \) are the angles from the i-th base station BS_i to V_k and T_j respectively, and \( w_0 \) is a constant for controlling the attraction force to repulsion force ratio. For instance, \( A_{54}^{4} \) in Fig. 5 represents the attraction force at the traffic units T_4 which is generated by the un-served units V_5.

If there are more than one un-served traffic units nearby, the resultant attraction forces \( \vec{A}_j \) for the traffic units T_j are calculated by vector sum for all the un-served traffic units V_k, expressed as (5).

\[
\vec{A}_j = \sum_{k=0}^{U_j} \vec{A}_{kj}^{\lambda},
\]

where \( U_j \) is the number of un-served traffic units near the i-th base station.

The new repulsion force \( \vec{R}_{ij}^{\lambda} \) during the λ-th iteration is then obtained by adding two parts together, the repulsion force from the previous iteration, and the projection of the resultant attraction force onto the radial direction. It is formulated as (6) in polar coordinates.

\[
\vec{R}_{ij}^{\lambda} = \left( |\vec{R}_{ij}^{\lambda-1}| + |\vec{A}_j^{\lambda}| \cdot \cos(\theta_j - \theta_{ij}), \theta_{ij} \right),
\]

where \( \vec{R}_{ij}^{\lambda-1} \) is the repulsion force from the previous control iteration, \( \theta_j \) is the angle of the resultant attraction force \( \vec{A}_j^{\lambda} \).

As shown in Fig. 5, the attraction forces from the un-served traffic unit V_5 increase the repulsion forces of those traffic units at the same side, and decrease for those at the other side. This has resulted in a distortion on the cell contour of BS0 so that V_5 could have enough repulsion force to be included in the contour.

The attraction force described here is different with that of bubbles, the main reason is we separate the local optimization (calculating cell contours) and inter-cell communication (re-adjusting cell contours) as two processes so that the analysis and evaluation can be simpler. However, both attraction forces share the same fact that cell contour or bubble surface tends to move towards the vacuums, which is the main purpose of employing attraction forces here. The attraction force in the bubble oscillation algorithm is used as a mean of inter-cell communication. The advantage of this approach is that it can make the iterative adjustments more directional and accurate. This is the main difference between our previous work, which uses cooperative negotiations as a generic inter-cell communication technique [17].

The new cell contours of the base stations BS_i are calculated using the new repulsion forces with the same local optimization method described before. If some served units (usually at the opposite directions) have to lose their service due to the capacity or power constraints, they will generate new attraction forces in the next iteration to attract adjacent cells. To avoid it directly oscillating back in the new iteration, any cell that has changed its coverage needs to wait for one iteration to be able to change its coverage again. This simple scheduling method provides the damping effect to this bubble oscillation algorithm, which reduces the oscillation amplitudes (the amount of cell contour changes) during the iterations.

This adjustment iteration is repeated until the optimization has converged or the maximum iteration number has reached. After that, the traffic hot-spots are dispersed through the whole network with this algorithm.

Only after the optimization converges or all the un-served traffic units have been eliminated, the cell contours are applied into the network. This means the network does not change its coverage until the optimization is done, so the oscillation described above does not cause any disturbance to the current network.

IV. Simulations and Results

Simulations are performed to test the performance of the approach. The results of using bubble oscillation algorithm

---

[Note: The image contained within the document is crucial for understanding the context of the described algorithm, especially the visual representation of the cell contours and the iterative process.]
are compared with the conventional ones, where base stations provide services only for the closest users by circular cellular coverage. In order to reduce the boundary effects, a 100 diamond-mesh model is used and the cell radius $R = 1$, as shown in Fig. 6. The hexagon network topology is used, as this is the most common one in network planning area. So the maximum number of base station that can provide service for a traffic units is fixed as $M = 3$.

A. Simulation Configuration

The approach is being evaluated using a sequence of 200 traffic snapshots taking from an imaginary mobile cellular network where some hotspots are forming. The time interval between two traffic snapshots is 60 seconds. The configuration for traffic snapshots is:

- Each traffic snapshot contains 50,000 users in the whole area;
- All the users are uniformly distributed at the first snapshot;
- One to ten traffic hotspots are forming from the first to the last snapshot;
- Each hotspots contains 2500 users, and the rest users are always uniformly distributed in the whole area;
- The locations of traffic hotspots are uniformly distributed with the minimum central distance of any two greater than $3R$;
- The locations of the users in hot spots follow normal distributions, whose mean value $\mu$ is the central point of each traffic hotspot, and standard deviation $\sigma = 0.5R$.
- The active and idle time for each user has a negative exponential distribution. The mean values are 120 sec and 720 sec respectively.
- One base station is situated in the centre of each cell, and each base station antenna has a capacity of serving 120 traffic units.

B. Simulation Results

Simulations have been performed for 40 traffic scenarios, and each of them contains 200 traffic snapshots. Due to the space limit, only three results are presented in Fig. 7. They
are the average system capacity for three groups of scenarios that have 1, 5 and 10 traffic hotspots for each group.

As the hotspots form, the system capacity of mobile network decreases. However, the system capacity of the scheme using geographic load balancing with the bubble oscillation algorithm is always better than the conventional one, especially when hotspots come into being. When there are more traffic hotspots, the performance improvements are more significant, since there is more space for optimization. The plateaus in the second and third graphs are caused by the directional movements of users, which is of no interest for our work. The results presented here is as accurate as our benchmark obtained with global optimizations, while it is much faster than that.

When the traffic is uniformly distributed, the optimized cells are nearly circular, and when hotspots present, the optimized cells are distorted to serve the hotspots. Some results are presented in Fig. 8. The big triangles represent the base stations, and the small dots represent traffic units. Traffic units with different colors mean that they are served by different base stations, and the mixed colors mean that they are shared by two or three base stations.

We also test this system with different numbers of users in hotspots, the simulation results for scenarios with 2000, 2500, and 3000 users in each hotspot are shown in Fig. 9.

When the hotspots are relatively light, the geographic load balancing system can almost avoid un-served traffic appearing, while the conventional network has lots of blocked or dropped users. When the hotspots are very heavy, the geographic load balancing system reaches its optimization limit, and its system capacity also decreases. However, it still performs much better than the conventional one.

The computation time of bubble oscillation algorithm for one traffic snapshot is about 8.3 seconds on average on a PC computer (Pentium III 750MHz CPU with 256MB RAM). This time has excluded the simulation time, which will be approximately the same for both conventional network and optimized network. In practice, this control algorithm will be running at base station controllers (BSCs) or radio network controllers (RNCs), which usually manage seven base stations each. Then it only takes about 0.57 seconds on average for each optimization cycle given the similar processing capability.

V. DISCUSSIONS

The simplest bubble oscillation algorithm has demonstrated very good accuracy and efficiency for the geographic load balancing in mobile network. However, there are still some issues that could be clarified.

The $w_0$ in (4) is used to control the attraction force to repulsion force ratio, which affects the speed of cell distortions. If $w_0$ is chosen too small, the un-served traffic units cannot provide enough force to attract the cell coverage, which results in slow convergence. As shown in Fig. 10, very small $w_0$ values, such as 0.01 and 0.05, slow down the optimization convergence, since more iterations are needed to generate enough attraction forces to change the cell size and shapes.

Fig. 8. Optimization results for (a) uniform traffic and (b) hotspot traffic. (Silent users are not plotted.)

For the $w_0$ values from 0.05 to 100, the convergence times for optimizations are almost identical. The large $w_0$ values do not affect the accuracy, which is usually the price of fast convergence. This is due to the self-correcting feature of the bubble oscillation. If a cell has been distorted to the wrong direction, the un-served traffic will probably appear at the other direction, and automatically pull the cell back. So this algorithm is very robust and the parameter tuning is rather easy, as long as $w_0$ is large enough to allow the start of cell reshaping. From our experiments, any value of $w_0$ greater than 0.3 (approximately equal to $0.3 \cdot | \bar{R}_{ij}^{0} |_{\text{average}}$) are good enough to produce sensible results.

Another refinement is the initial values for the radial re-
pulsion forces. Since the traffic changes in the real world are always continuous, it is reasonable to use the initial values of radial repulsion forces from the last traffic control cycle. Apart from this, the equation of initial repulsion forces values can also be refined to speed up the convergence. Fig. 11 shows an example of using refined equation for calculating the initial values so as to improve the results for both first iteration and convergence. Instead of use (3), the initial repulsion forces are calculated by (7)
\[
\vec{R}_{ij}^0 = (w_1 L_{ij} - d_{ij}, \theta_{ij}),
\]
where \(w_1\) is the weight, and \(L_{ij}\) is the total traffic of a 60 degree sector where traffic units \(T_j\) is inside.

\(L_{ij}\) is used to estimate the traffic condition of the adjacent base station since the sector has large overlapping area with the adjacent cell. If the adjacent one is heavily loaded, the initial repulsion force \(\vec{R}_{ij}^0\) can be intentionally increased to reflect the fact that base station \(BS_i\) has more obligations to serve the users in the sector towards the heavily loaded one.

VI. CONCLUSIONS

This paper has investigated a novel bubble oscillation algorithm for geographic load balancing in mobile cellular network. The system capacity is improved by adjusting the cell size and shape according to the current geographic traffic distribution. The near optimum cellular coverage is obtained using the bubble oscillation algorithm. The geographic load balancing is performed by emulating the bubble oscillations, as the process of base stations re-allocating un-served traffic units is very similar to that of bubbles’ filling the vacuums between them. The results from computer simulations have demonstrated the benefits of the geographic load balancing system as well as the efficiency and accuracy of the algorithm.

The bubble oscillation algorithm described here can be used for the geographic load balancing system at other environments or other similar multidimensional resource allocation problems.

Future work will investigate the effect of location accuracy, system robustness, and create better simulations with richer world models and scenarios.

ACKNOWLEDGEMENTS

This work was supported through the US Office of Naval Research grant BAA 03-001 Provision of Quality of Service on Wireless Networks, grant ref N00014-03-1-0323.

REFERENCES


[19] 3GPP TS 04.35, “Location services (LCS); broadcast network assistance for enhanced observed time difference (E-OTD) and global positioning system (GPS) positioning methods,” http://www.3gpp.org, May 2002.