# Using Geo-Spatial session tagging for smart Multicast session discovery

Piyush Harsh*
Computer & Information Science & Engineering
E301 CSE Building
University of Florida, Gainesville, FL, USA
pharsh@cise.ufl.edu

Richard Newman†
Computer & Information Science & Engineering
E301 CSE Building
University of Florida, Gainesville, Florida 32611
nemo@cise.ufl.edu

## ABSTRACT

IP multicast is increasingly seen as efficient mode of live content distribution in the Internet to significantly large subscriber bases. Despite its numerous benefits over IP unicast, multicast has not seen widespread deployment over modern networks. Network complexity and session discovery issues have plagued IP multicast since its inception. The Internet research community is in general agreement to move over to SSM (Source Specific Multicast).

With IGMP v 3 (Internet Group Management Protocol) and SSM, the source discovery burden will rest with the end user. Channel discovery is one of the few stumbling blocks remaining to be solved for successful and widespread deployment of multicast. In an earlier work a DNS (Domain Name System) aware multicast session discovery architecture, mDNS, has been proposed which is distributed, hierarchical and globally scalable.

In this paper authors propose to leverage mDNS architecture by enabling multicast sessions to be tagged using geographical and spatial information based on the channel contents or service provider location. They further propose automatic geo-coding of session registration information as the content provider registers session information with mDNS. They also provide necessary design changes and give data models and data structures to support seamless location sensitive session retrieval as part of search query results to be furnished to the end user. They have also envisaged scenarios in which geo-tagging would enhance end user experience and would enable smarter query result generation.

## Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design

*PhD Candidate at CISE, University of Florida

†Assistant Professor at CISE, University of Florida



## General Terms

IP Multicast

## Keywords

source discovery, geocoding, session directory, mDNS, multicast

## 1. INTRODUCTION

Consider a scenario where you live in a particular part of a country and love listening to jazz music. You travel often and while on the go you would love to continue listening to your favorite genre of music. You perform a geo-location sensitive search to find a multicast[4] session that broadcasts jazz music from a location near you.

Picture another scenario, emergency channels multicasted from a region hit by a natural disaster would generally be more effective in providing real time relief information to residents in that area. Information such as where to go to get clean drinking water, a bag of ice and medical aid or to disseminate casualty information can be updated in real time by emergency workers present at disaster sites rather than someone sitting at a far away location. Geo-tagging such multicast sessions would help people discover relevant sessions faster and with high accuracy. In fact in some cases, multicast sessions could also foster better interagency coordination enabling them to orchestrate an efficient relief program in the affected areas.

Multicast can also be used to effectively push out weather and safety alerts specific to a given locality. A subscription based multicast session channel can be established. Its session details along with geographical location can allow interested users to locate the session and subscribe to it. This can be very effectively utilized in sending out geo-specific alerts in real time.

Geo-tagged multicast sessions could also herald an era of real-time yet discoverable citizen news reporting by eye witnesses at news sites. Consider a scenario where a major traffic pileup has occurred on a major US highway I95, a few eye witnesses on accident site may start a live video feed using their camera phones (considering modern cell phones are packing in more and more compute power), using 3G[2] or GPRS (General Packet Radio Service)[8], register the multicast session using descriptive keywords such as, I95, pileup, accident etc. and let the whole world watch the news as it unfolds.

Raging California wild fires have made the county officials issue voluntary evacuations. Homeowners who decide to move out are always on their toes to find out the status of their homes. A few daredevils who decide to stay back, could start a video feed of their surrounding, geo-tagging their session with relevant location would make such sessions discoverable with more accuracy and homeowners who vacated could find the status in their areas.

Furthermore network traffic if sourced from nearby location generally is more reliable and impervious to network vagaries. Link capacities and traffic profile have a tremendous impact on the quality of sessions that have a larger hop count. Therefore usually one would want to get contents from sessions hosted from a location near oneself.

These are a few scenarios among many that suggest geo-tagging of multicast sessions could have significant impact on the way people would consider using multicast in the future. Not only will multicast be attractive to ISPs (Internet Service Providers) for efficiency reasons, but with increased usability and capabilities, it will be more appealing to the public, creating demand. This interest and the capabilities will in turn enable various new services to be offered, a few that are envisioned above, that could not be offered beforehand.

Researchers might argue, why not leverage existing search engines like Yahoo© and Google© in few of the above mentioned scenarios. Table 1 shows the categories into which multicast sessions can be classified. Clearly due to latency

| | Planned | Ad-Hoc |
|---|---|---|
| Transient | | mDNS |
| Persistent | | |

**Table 1: Multicast session categories**

issues, search engines can not be used in cases where multicast sessions are transient and ad-hoc. Geo-tagging of sessions and a structured multicast session discovery architecture that allows for real-time session discovery would go a long way in popularizing multicast.

IP Unicast addresses are assigned to regional ISPs en-block. This allows for geographical information to be automatically gathered based on the unicast IP from which the client is connecting. This has already allowed for region specific, targeted, online advertisements to be placed alongside contents sent to remote clients. Internet yellow pages, although it usually requires valid businesses to register with such services, could also apply internet search engine technology to compile a more comprehensive local businesses and services database (although not always because the business portal could be hosted at some far away location).

IP Multicast does not enjoy such benefits. Multicast addresses belong to a common pool with no long term allotment and ownership attached to such addresses. An interesting question here is - Can a distributed architecture be built that would allow multicast sessions to be geographically tagged? Further, even if such a service exists, what benefits will it bring to the users? Will such geographical tagging add any real value to sessions disseminated using multicast? Would the proposed architecture make sessions discoverable in real time. Given the transient nature of most multicast sessions, this last problem if solved could push multicast into mainstream use.

In this paper the authors have tried to answer such questions. mDNS[6], which is a DNS aware, hierarchical multicast session directory architecture has already been proposed. mDNS working in tandem with HOMA[7] (Hybrid Overlay Multicast Address allocation scheme) provides for efficient use of IP multicast (even for IPv4) and also makes sessions discoverable in real time. In the next few sections they propose mechanisms to add geographical tagging of multicast sessions, and a preliminary data structure to be added in the mDNS architecture. They analyze the benefits and drawbacks of such a scheme and envision scenarios and services where geo-tagging of sessions will prove valuable and enhance multicast appeal to the general public.

## 2. TOWARDS GEO-SPECIFIC SESSIONS RETRIEVAL

In this section authors will describe their proposal of a system whose addition to current mDNS architecture would enable clients to retrieve multicast sessions based on geographical parameters specified during search initiation.

### 2.1 mDNS - Brief introduction

mDNS[6] which is a hierarchical multicast sessions directory architecture has already been proposed by researchers. Their proposal uses existing DNS[13] infrastructure to locate the appropriate MSD (Multicast Session Directory) server in the target domain and then the MSD server locates session details and returns them back to the querying client enabling it to join the desired multicast session. Their proposal makes use of a URS (URL Registration Server) under each domain in order to enable users to access session details via standard URLs (Uniform Resource Locators) (see below).

mDNS also allows end users to perform both domain specific multicast session search as well as global search. It is the responsibility of the session creator or content provider to register appropriate session details with MSD servers. The addition of MCAST records in every DNS server allows mDNS URLs to function correctly. Below is a sample MCAST[1] [2] record entry in a DNS server.

```
@MCAST{
    ANYCAST=a.b.c.d
    CMCAST=233.[ASN Byte1].[ASN Byte2].XXX
    PMCAST=233.[ASN Byte1].[ASN Byte2].???
    URS=x.y.z.w
}
```

No port numbers are included in an MCAST record keeping in line with typical DNS record format. mDNS proposers have assumed that those will be well known port numbers and possibly IANA assigned for standardization purpose. A typical mDNS URL could be: **mcast.cs.abc.edu/netsec**. This URL points to **MCAST** record in the DNS server at **cs.abc.edu** administrative domain. Once this record is read, the browser (or any similar tool) can discover the anycast address of the MSD server(s) in CS domain and contact it directly for session details of that particular multicast session which registered the domain unique keyword **netsec**

[1]ANYCAST refers to anycast IP address[9] used to address multiple MSD servers maintained under a given domain, CMCAST and PMCAST refers to GLOP multicast addresses[11] and are used in constructing hierarchy in mDNS
[2]ASN refers to Autonomous System Number

with the URS server at CS. Figure 1 shows the typical mDNS hierarchy once mDNS bootstrapping process is over[3].

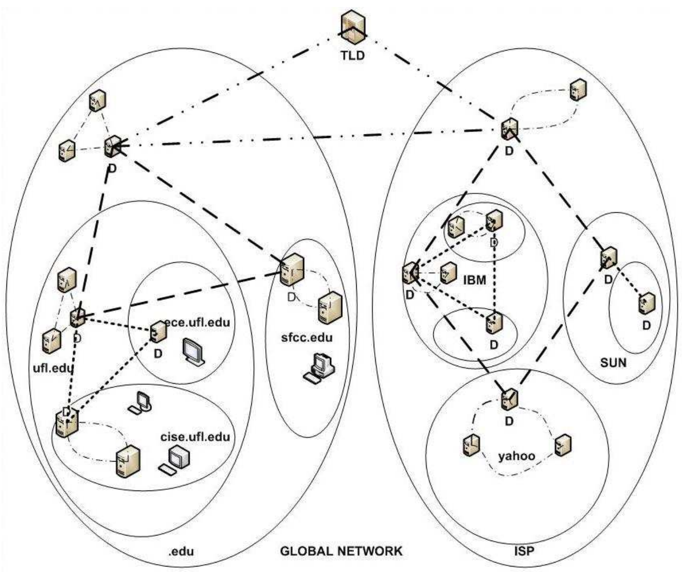


**Figure 1: Typical mDNS hierarchy**

## 2.2 Leveraging mDNS: geo-tagging

Every location on earth can be addressed using a set of coordinates, e.g. latitudes and longitudes. Latitudes range from 0°to +90°in the northern hemisphere and 0°to -90°in the southern hemisphere. And longitudes ranges from 0°to +180°in the eastern hemisphere and similarly from 0°to -180°in western hemisphere. Figure 2 shows this division. In

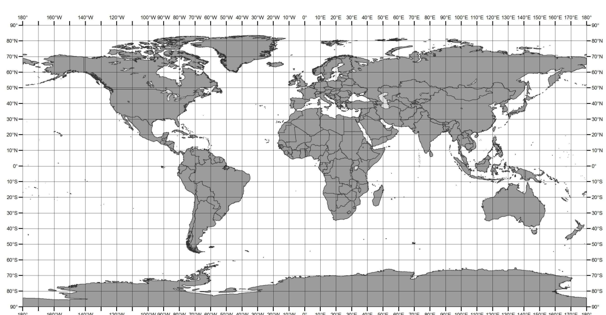

**Figure 2: world coordinate system**

this coordinate system, each latitudinal second distance is 30.82 meters apart. That is each latitude degree separation is 110.9 km in length. Now because the longitudinal lines meet at the poles, and further due to earth's not so spherical structure, the linear distance between longitudes are not so straightforward. It can be calculated using the following mathematical formula[1] -

$$\frac{\pi}{180^\circ} \times \cos\phi \times \sqrt{\frac{a^4\cos(\phi)^2 + b^4\sin(\phi)^2}{(a\cos\phi)^2 + (b\sin\phi)^2}} \qquad (1)$$

at any given latitude $\phi$ and using $a = 6,378,137m$ and $b = 6,356,752.3m$ for every degree longitudinal change.

[3]TLD refers to Top Level Domain

In this paper authors are proposing mDNS registration process to include auto geocoding. Content providers and session creators, in mDNS, during registration process must furnish a valid physical address where they are located or what specific area (if any) the multicast content is most significant about. It is proposed to use Google© Maps API[5][10] for translating any valid physical address into corresponding latitude and longitude values. It is then possible to use equation (1) and geographical coordinates associated with multicast sessions in many interesting ways.

# 3. SYSTEM DESIGN

There are several candidate data structures that authors have analyzed for use in mDNS database. Let us divide the planet into grid structure. Major lines of the grid being the latitudes and longitudes, each being 1°apart. Therefore under this proposal, the whole planet will be mapped onto a 180x360 grid space.

Another significant observation to be made is the water coverage. Since about 70% of earth's surface is covered by water[4], it would not be unreasonable to estimate that around 70% of grid area will map these water bodies. Furthermore vast areas of landmass is also uninhabited including dense forests, deserts and mountainous areas. According to the article published by Conservation International, "Global analysis finds nearly half the Earth is still wilderness"[5] [12], thus grids in the grid map that does not lie over water-bodies, an additional 50% of such grids will map areas uninhabited/sparsely habited by mankind.

Keeping all the above data in mind it would be only reasonable to expect the grid map to be sparsely populated with only those grids being possibly used that lies above populated zones with significant internet presence.

## 3.1 A sparse-matrix-linked-list cross referenced database proposal

The figure 3 shows the fundamental idea behind the proposed data structure. Authors have already justified why the planetary scale grid (or level 0 structure) is best implemented using a sparse matrix[14] data structure. Only those grid positions are maintained that have multicast session entry whose geo-coordinates lies within the corresponding coordinates for that grid location. But since a 1°x 1°grid may represent an area as large as 111.3 x 110.9 $km^2$ (on the equatorial plane), it would be appropriate to further subdivide the region into smaller grids. In figure 3 the case where each grid is subdivided into 2x2 smaller grids is shown, but the choice of number of divisions would depend on number of factors, such as, depth of tree desired, areal resolution, etc.

As an example if the final areal resolution desired is, say, 5 km x 5 km, then using a branching factor of 2 x 2 grids, this could be achieved at equatorial plane by a tree of height 4 or 5 levels (where the planetary grid is assumed level 0). A tree of height 4 would result in areal resolution of 6.95 km x 6.93 km and that of height 5 would result in areal resolution of 3.48 km x 3.47 km at the equatorial plane.

[4]source: Retrieved July 31, 2008, from `http://www.sciencedaily.com/releases/2003/08/030826065131.htm`

[5]source: Retrieved October 17, 2008, from `http://www.eurekalert.org/pub_releases/2002-12/ci-gaf120202.php`

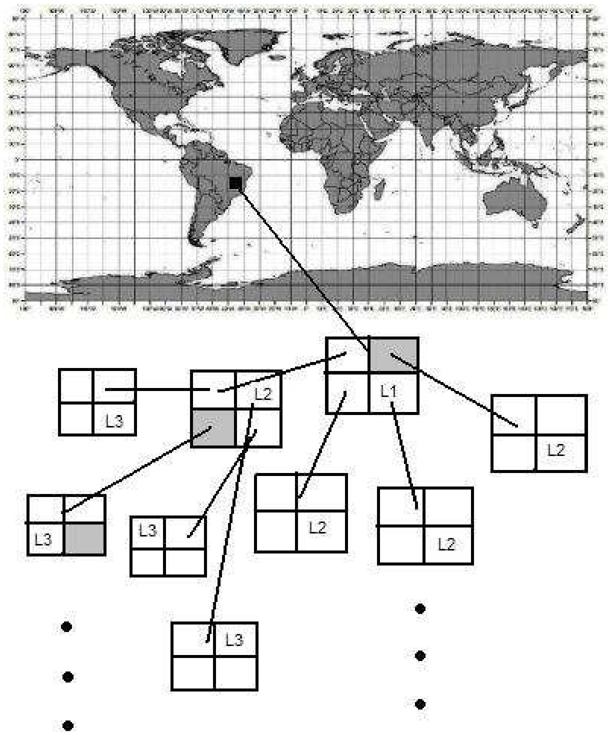


**Figure 3: Proposed data structure - Sparse Matrix and 4-ary Tree**

Since the distance between longitudes vary with latitudes, the east-west resolution with each additional tree depth will vary and this variation will be governed by equation 1.

Let **K** be the branching factor (i.e. each grid would be subdivided in k x k grids), then -

$$\frac{110.9}{K^n} km \qquad (2)$$

$$\frac{\pi}{180^\circ} \times \cos\phi \times \sqrt{\frac{a^4\cos(\phi)^2 + b^4\sin(\phi)^2}{(a\cos\phi)^2 + (b\sin\phi)^2}} \times \frac{1}{K^n} km \qquad (3)$$

equation (2) governs the north-south grid resolution at tree level **n** at any coordinate and equation (3) governs the east-west grid resolution at tree level **n** at latitude $\phi^\circ$. At the lowermost level, the grid maintains a pointer to a linked list or some other appropriate data structure containing multicast session records that fall under its coordinate range.

## 3.2 Incorporating into mDNS: the Specifics

Under mDNS current architecture, each MSD (Multicast Session Directory) server in a domain maintains session parameters of various multicast sessions whose registration originates from within that domain (as shown below).

```
@MSD_SESSION_INFO
    {
        MULTICAST IP
        MULTICAST PORT
        URS REGISTERED KEYWORD
        KEYWORD LIST
        SESSION TYPE
        SESSION SCOPE
        START TIME
        END TIME
        LEASE DURATION
    }
```

Each mDNS MSD server maintains several databases cross-referenced with each other to enable optimal session data retrieval and in order to enable various functionality supported by mDNS architecture such as global and local searches and URL based data retrieval to name a few. With the addition of sparse-matrix corresponding to planet grid map, authors hope to enable searches based on topological and geographical constraints and criteria.

After addition of geocoding capabilities to MSD servers, authors propose to augment each session record with geo-spatial data such as planetary coordinates and city location fields. It is important to note that these are strongly linked with the nature of content being sent on that channel and not necessarily the physical location of the server from where they are being broadcasted. Their intuition suggests these will be, in most cases, same. The expanded MSD session record will look somewhat as shown below -

```
@MSD_SESSION_INFO
    {
        MULTICAST IP
        MULTICAST PORT
        URS REGISTERED KEYWORD
        KEYWORD LIST
        SESSION TYPE
        SESSION SCOPE
        START TIME
        END TIME
        LEASE DURATION
        GEOLOCATION-LATITUDE
        GEOLOCATION-LONGITUDE
        GEOLOCATION-COMMON-NAME
    }
```

Whenever a session creator registers a session with appropriate MSD server in its domain, the MSD server not only creates a MSD_SESSION_INFO record it its original proposed database for that session, but also cross-links that entry with its corresponding geo-spatial grid location in the planetary sparse matrix data structure. In the original mDNS implementation, the URS servers maintain two data structures, one hash-table for quick retrieval of registered URS keywords, and second a min-priority queue where entries are sorted according to lease expiration time of URS keywords.

When any URS entry is removed from priority-queue, the corresponding entry is also removed from the URS hash-table. A similar mechanism is also built into MSD servers; authors propose to link the deletion behavior between MSD session database and the sparse matrix grid data structure. Once a record is removed from the MSD records database because of lease expiry, the cross-linked entry is deleted from the sparse matrix grid structure as well.

## 3.3 Incorporating into mDNS: search example

Figure 4 shows a typical mDNS administrative domain network components. This subsection will explain how geo-tagging provides an additional search dimension through an example search query. Maintaining the sparse matrix coordinates grid allows end users to perform faster geo-spatial sensitive searches.

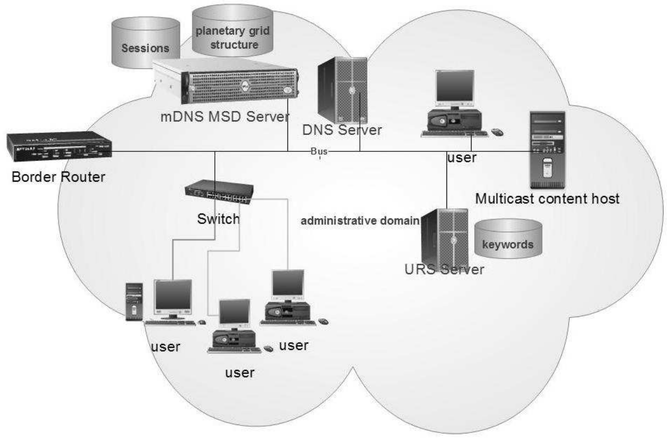


**Figure 4: a typical administrative domain in mDNS architecture**

Imagine a user interested in finding out multicast sessions with contents related to jazz within 10 miles of downtown Miami. Using Google© Maps API and its Java wrapper class, mDNS search module translates “downtown, miami” into corresponding latitude and longitude values. The mDNS hierarchical structure and the global search algorithm[6] locates the MSD server that maintains all global session records for keyword ”jazz” and passes on the search query to that MSD server. If the search scope is local, the query does not go beyond the MSD server hosted in the querist domain. Remember a domain specific search too goes directly to the MSD server located in the target search domain and not to MSD server responsible for search keyword for all sessions globally (these two servers could coincidentally be the same). A search query would look somewhat as shown below -

```
@MSD_SEARCH_DATA
    {
        KEYWORD-LIST
        GEOLOCATION-LATITUDE
        GEOLOCATION-LONGITUDE
        SEARCH-PERIMETER
        CONTENT-TYPE
        CONTENT-SCOPE
    }
```

In the above structure, content type could be either multimedia, text, conference, whiteboard to name a few. Search perimeter denotes the radius of search from the specified latitude and longitude values. Content scope could be either global or local. Keyword list is the only mandatory field; all other fields could be absent. If latitude and longitude values are absent then search-perimeter value is deemed useless. Figure 5 shows the user interface of mDNS multicast content browser. It can handle valid mDNS URLs. It will be used by end users and will be fully capable in processing and rendering different types of multicast sessions, additionally it will enable user to preform session searches. The browser itself is part of the mDNS suite which is a work in progress.

[6]the global search algorithm for mDNS has undergone significant revisions from its original form, details are outside the scope of this paper

Once the search query reaches the appropriate MSD server, if search-perimeter and coordinates are specified, it locates the corresponding grid position in the sparse matrix structure and descends down to appropriate level and grid position depending on the search perimeter, extracts the cross-linked session entries and further filters the result against the specified keyword list.

If multiple keywords are present in the query, it is very likely that the query would go to different MSD servers responsible for each of the specified keyword. The results from each such MSD server is collected locally at the requesting node and union or intersection set operations are performed depending on keyword operators OR / AND.

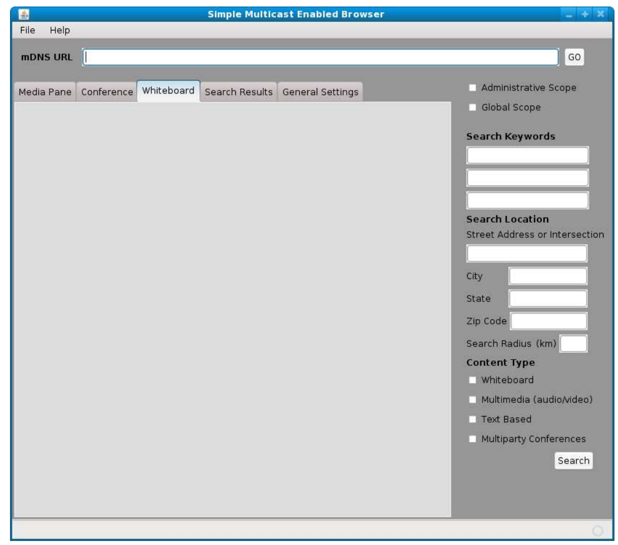


**Figure 5: mDNS multicast browser for end user**

# 4. GEO-TAGGED mDNS SEARCH: COMPLEXITY ANALYSIS

The complexity of locating the multicast session records in mDNS augmented with additional sparse matrix grid coordinate data structure depends on several factors such as sparse matrix representation format, the grid branching factor **k**, the search radius **r** requested, and the system areal resolution desired **d** km.

Since it is not known beforehand which grid will be populated (or used), a linked list representation for the (sparse matrix) grid seems appropriate. Given a coordinate value, the row index is computed in $O(1)$ time. To find appropriate column index one must traverse the linked list, whose length is also bounded above by the value 360 (for longitudes between -180°to +180°). Therefore the total complexity to locate the appropriate grid in linked list sparse matrix representation is $O(1)$.

The maximum depth **h** of the **k x k** grid tree rooted at the linked list node corresponding to any coordinate pair is governed by equation $d \geq \frac{110.9}{K^h}$ or $h = \lceil log_k \frac{110.9}{d} \rceil$. Since the actual multicast session records are maintained only at the leaf nodes in this tree, regardless of the search radius **r**, one must traverse the whole depth of the tree.

The depth of the tree **h'** where each grid height is greater or equal to the search diameter **2r** can be found using equation $h' = \lfloor log_k \frac{110.9}{2 \times r} \rfloor$. At this depth, each grid's length can be found out using equation (3) by substituting **n** by $h'$ in that equation. Now number of lateral (east-west) grids **N** that one must cover in order to compensate for decrease in longitudinal distance with proximity to poles can be found out using

$$N = \lceil \frac{2 \times r}{\frac{\pi}{180^\circ} \times \cos\phi \times \sqrt{\frac{a^4 \cos(\phi)^2 + b^4 \sin(\phi)^2}{(a\cos\phi)^2 + (b \sin\phi)^2}} \times \frac{1}{K^{h'}}} \rceil \quad (4)$$

Total number of tree leaf grids that one may possibly have to traverse to cover the search radius can now be easily calculated using $N_{leafgrids} = N \times k^{h-h'}$

Once a leaf grid is reached, the system then traverses the linked list in its entirety to get the set of candidate multicast session records that satisfy the search geo-spatial radius criteria. This operation depends on the length of the corresponding linked list at each such candidate leaf grid cell location.

Hence the overall worst case time complexity for enabling geo-spatial search under modified mDNS architecture can be approximated by -

$$C \times (N \times k^{\lceil log_k \frac{110.9}{d} \rceil - \lfloor log_k \frac{110.9}{2 \times r} \rfloor} \times O(list)) \quad (5)$$

where C is a constant that can vary between 1 and 4 depending on the proximity of the search query coordinates to the grid's upper or lower edges or corners. $O(list)$ is the complexity of linked list traversal for linked-list maintained at each leaf grid cells which depends on the length of the corresponding linked lists. This complexity could further be reduced if the leaf linked-lists are replaced by hash tables and using perfect hashing functions[3].

# 5. STATUS AND CONCLUSION

mDNS architecture is extremely conducive to partial / incremental deployment. For domain specific search, the full hierarchy need not exist. Even if there are disconnected mDNS islands in the Internet, they would continue to be operational and would enable real time session discovery to users within that zone.

In this paper authors presented a mechanism to leverage existing mDNS architecture by adding geocoding capabilities into the MSD server. This change allows end users to perform multicast session search based on his/her geographical preferences in addition to usual keywords search capability that mDNS supports. They believe this geo-spatial augmentation of session search will enable end users to perform smarter search and would provide additional service capability into IP multicast. One such service could be disaster management information broadcast.

Authors also presented worst case complexity analysis of current proposed approach. Their hope is that this proposal would make multicast more appealing to the public and that in turn would provide ISPs incentives to deploy native IP multicast in their networks.

Components of mDNS architecture including URS server, MSD server, multicast browser for end users, multicast content management suite for content creators are at various stages of development. Authors are attempting to integrate changes proposed in this paper into mDNS development cycle without causing major disruptions in overall project development cycle.

# 6. ACKNOWLEDGMENTS

Authors are grateful to fellow student researchers in CISE CONS Lab notably InKwan Yu, Mahendra Kumar and Panoat Chuchaisri for their constructive criticism of their current and past work. Also they appreciate the time that Dr. Randy Chow of CISE has spent reviewing this and past research papers for them. They extend special thanks to Dr. Grant Thrall of College of Liberal Arts and Sciences at University of Florida for his help in verifying some geographical facts presented in this paper.

# 7. REFERENCES

[1] *A guide to coordinate systems in Great Britain - An introduction to mapping coordinate systems and the use of GPS datasets with Ordnance Survey mapping.* Ordnance Survey, March 2008.

[2] G. Camarillo and M. A. Garcia-Martin. *The 3G IP Multimedia Subsystem (IMS): Merging the Internet and the Cellular Worlds.* John Wiley & Sons, 2006.

[3] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms.* The MIT Press, 2001.

[4] S. E. Deering. Multicast routing in internetworks and extended lans. In *SIGCOMM '88: Symposium proceedings on Communications architectures and protocols*, pages 55–64, New York, NY, USA, 1988. ACM.

[5] Google. Google MAP api. http://code.google.com/apis/maps/.

[6] P. Harsh and R. Newman. mDNS - A proposal for hierarchical multicast session directory architecture. In *Proceedings of the 2008 International Conference on Internet Computing*, pages 31–37, 2008.

[7] P. Harsh and R. E. Newman. An overlay solution to ip-multicast address collision prevention. *IASTED EuroIMSA Conference Procedings*, Mar 17–19 2008.

[8] R. Kalden, I. Meirick, and M. Meyer. Wireless internet access based on gprs, 2000.

[9] D. Katabi and J. Wroclawski. A framework for scalable global ip-anycast (gia). *SIGCOMM Comput. Commun. Rev.*, 31(2 supplement):186–219, 2001.

[10] J. Liang. *JGeocoder - Free Java Geocoder.* http://jgeocoder.sourceforge.net/index.html.

[11] D. Meyer and P. Lothberg. *GLOP Addressing in 233/8*, 2001. Internet Engineering Task Force, RFC 3180.

[12] R. A. Mittermeier, C. G. Mittermeier, P. R. Gil, and J. Pilgrim. *Wilderness: Earth's Last Wild Places.* Conservation International, 2003.

[13] P. Mockapetris and K. J. Dunlap. Development of the domain name system. *SIGCOMM Comput. Commun. Rev.*, 18(4):123–133, 1988.

[14] M. Silva. Sparse matrix storage revisited. In *CF '05: Proceedings of the 2nd conference on Computing frontiers*, pages 230–235, New York, NY, USA, 2005. ACM.