A Survey of Service Discovery Protocols in Multihop Mobile Ad Hoc Networks

An in-depth review of service discovery protocols (SDPs) in multihop mobile ad hoc networks analyzes SDP building blocks and determines SDP suitability based on architecture, mobility, and network size.

Mobile ad hoc networks (manets) include a variety of devices, such as cell phones, PDAs, laptops, and other relatively larger devices. These devices can move at high or low speeds or even remain stationary, entering and leaving the system when switched on or off. Such a variety of devices also offers a variety of services. A service is any tangible or intangible facility a device provides that can be useful for any other device. Services comprise those for software and hardware. A software service, for example, can be a simple file, such as an MP3 file, or a software implementation of an algorithm, such as converting one audio file format to another. A hardware service, for example, can be a printer that a mobile device can use wirelessly. To benefit from these services, a device must be able to both locate them in the network and invoke them. Service discovery protocols (SDPs) enable these capabilities.

SDPs play a role in wired networks, single-hop wireless networks, and multihop manets. In wired networks, the many industry standard SDPs include Sun’s Jini, Microsoft’s Universal Plug and Play, IBM’s Salutation, and the Internet Engineering Task Force’s Service Location Protocol. In single-hop wireless networks, SDPs include DEAPSpace and the industry standard Bluetooth SDP. Wired networks have stationary devices, whereas single-hop wireless ad hoc networks have restricted mobility with a low rate of devices joining and leaving.

This survey focuses on multihop manet SDPs, one of the many approaches to realizing pervasive computing environments. In these networks, devices can have unrestricted mobility—that is, no constraints on joining or leaving the network—making service discovery difficult. (SDPs for wireless sensor networks, such as the ZigBee standard SDP, are out of our work’s scope, because such SDPs focus on energy optimization rather than mobility.)

Most SDPs in wired networks work at the application level. In theory, any such application-level SDPs can work for manets as well. In practice, however, mobility and resource scarcity introduce two dimensions that SDPs for wired or single-hop wireless networks don’t take into account: location awareness and physical proximity between the service provider and the user. Thus, physical proximity remains undetectable. In multihop manets, the service provider and user’s physical proximity is a factor. Using
bandwidth efficiently and reducing link breakages requires services that are physically close and at fewer hops. The Open Systems Interconnection architecture layering approach works well in wired networks without resource constraints, but it introduces inefficiencies in resource-constrained manets. Cross-layering can improve efficiency, for example, in a specially designed SDP that integrates service discovery with routing. Many proposals have focused on multihop manet SDPs, but none are as developed as those for wired or single-hop ad hoc networks.

Several comparative studies cover SDPs in all three types of networks, such as those by Chunglæ Cho and Duckki Lee; Feng Zhu, Matt Mutka, and Lionel Ni; and Raluca Marin-Perianu, Pieter Hartel, and Hans Scholten. However, none closely analyzes only SDPs for multihop manets. We selected 12 multihop manet SDPs, identified the basic building blocks of SDPs in general, and analyzed the selected SDPs in the context of these building blocks.

Service Discovery Protocol Framework

Figure 1 shows the basic building blocks of a manet SDP framework: description, registration, discovery (comprising search and selection), routing, and mobility support. We’ll explain each of these components in detail later in this section, but, first, we’ll provide an overview.

A user interacts with the SDP by first initiating service description, registration, or discovery. (By “user,” we mean either an end user or a software application that acts on the end user’s behalf.) The service description describes the service in a format usable for registration or search. A proper description makes searching the service easier. After accessing the service description, the user can choose to search for the service or to register it. If the service is new, the user must register it—that is, store the service description in a location—so that it’s available for later discovery. Once the service is registered, users can find it through discovery.

The user interacts with the discovery mechanism, specifying the service through service description and selection criteria. The search component searches for the service and, when it finds all instances, the selection mechanism chooses the most suitable service provider. The separation between search and selection mechanisms is often only logical. For example, if the user wants a printer service, the SDP first will search for all the printers and then select one according to some criteria. In this case, the search and selection phases are independent. In other cases search and selection might occur simultaneously, meaning that search continues until the SDP finds a suitable printer.

SDP can also form overlay networks. An overlay network, which sits on top of the existing network, is a virtual network comprising a subset of nodes from the existing network and logical links between these nodes. The user selects the nodes to form a topology with special properties. A multihop protocol creates links between these nodes. An overlay network directs traffic to specific nodes, thus greatly reducing network traffic. Registration and discovery mechanisms can use overlay networks to help route the registration or search messages to selected nodes.

In manets, the network’s topology changes because nodes are mobile. If an SDP depends on a specific overlay topology, it must maintain the overlay, which might become faulty because of node mobility. So, to compensate for node mobility, the SDP includes a mobility support component.

**Service Description**

Service description is an abstraction of a service’s facilities and characteristics. A service requires a description to
enable other services or devices to use it. The nodes search for services only by looking at service descriptions the service provider advertises. A service without a proper description can remain completely unknown to other network devices. Service description includes service properties (such as service ID), server properties (such as load on a service provider), and network properties (such as maximum number of hops from the current host to a service provider).

is, service advertisements flood the whole network. In this case, the service searches only the local cache. Another approach is to store service descriptions in a subset of nodes by distributing advertisements to $n$ hops$^{5,14}$ (the advertisement diameter) or distributing advertisements to multicast groups of nodes.$^{7}$ By restricting advertisement flow, the protocols reduce memory requirements and increase the probability of discovering a service in the service provider’s vicinity. When service de-
to directory nodes. Depending on the protocol, these directory nodes might be preassigned the task of storing service descriptions, or the SDP might select them at runtime. The clients search only these directory nodes. For example in Service Rings, the protocol searches only the service rings’ service access point (SAP) nodes. In Kozat and Tassiulas’ and Jerry Tyan and Qusay Mahmoud’s work, the client queries only the virtual backbone nodes (VBNs) and each cell’s gateway, respectively. In Lanes, SDP nodes group together to form an overlay network that creates lanes of nodes. Each group is called a lane, and nodes in the same lane have the same directory replicated in each node cache. So, search uses unicast on any of these directory nodes in a lane overlay instead of on one specific node.

Selection. The query request from a client node to the network can result in many responses from matching services. Selection from among the discovered services is then essential to invoke one of the services. Selection can be done manually by the user or automatically, using a criteria-based algorithm. The criteria can be the service provider’s lowest hop count, current load, or velocity, or the available bandwidth of the communication channel between the service provider and the client.

Search and selection are often integrated. Alex Varshavsky, Bradley Reid, and Eyal de Lara demonstrated that proper integration improves overall network performance by localizing network communication, thus reducing interference and allowing multiple concurrent transmissions in different parts of the network. In Vincent Lenders, Martin May, and Bernhard Plattner’s work, the client selects services using two metrics: the number of hops from the service provider and the service’s capacity. The service selection algorithm is distributed and doesn’t involve direct interaction with the client.

Many manet SDPs simply ignore the selection issue.

Routing
The SDPs that register or search in all of the network’s nodes flood query or registration messages. An SDP can scale well if the network traffic it generates is limited. To limit network traffic, SDPs employ different routing mechanisms. For example, SDPs limit the flooding to hops or use multicast routing to specially formed groups of service providers, selective forwarding based on ontology, or selective forwarding based on the notion of potential.

SDPs can also form overlay networks and have specific routing mechanisms within these overlay networks. For example, the Service Rings protocol shown in Figure 2a forms an overlay structure, called a service ring, by grouping nodes that are physically close and offer similar services. Each service ring has a designated SAP that keeps the directory and has information about available services in the service rings. The SAPs are connected with other service rings’ SAPs. When a node wants to search for a service, the query is routed through the ring structure, passing through SAPs of other rings and reaching only those subrings that can possibly offer the service.

Kozat and Tassiulas’ SDP forms the virtual backbone for the VBNs, as Figure 2b shows. These VBNs constitute a dominating set in which all of the network’s nodes are either in this set or only one hop away from at least one member of the set. The VBNs...
keep the directory, which stores the advertised information about other services in the network. The client forwards the service request to the VBNs, which in turn multicast the query message to all the other VBNs. Multicasting the backbone nodes, rather than flooding all nodes in the network, reduces the overhead of broadcasting a query. The arrows in Figure 2 indicate the routing path of query or registration messages. Tyan and Mahmoud’s SDP divides the environment in hexagonal cells, with each cell having a gateway node. Each cell’s gateway node provides the directory services. When a client wants to search for a service, the query routes to only the gateway nodes. In Lanes, each node in the lane has the same copy of the directory. The registration or search messages travel through lanes by anycast routing.

**Mobility Support**

In manets, nodes frequently move and change their positions with respect to each other. In SDPs without directory nodes, all nodes keep information of only their own services. In such a case, the SDPs automatically handle mobility reselection to handle mobility. In re-discovery, SDPs probe the network for up-to-date information about the available service providers. In reselection, the SDPs select services based on only the current entries in the service table. Another method for handling mobility is reducing the advertisement diameter and advertisement time interval to compensate for the effects of rapidly changing vicinities.

SDPs that form overlay networks have special algorithms for maintaining the overlay structure, which becomes faulty due to mobility and logging in and out of nodes from the network. Service rings have special algorithms to maintain the ring overlay’s consistency. Each ring member knows only its successor and its predecessor. The appropriate SAPs periodically initiate RingCheck messages. The messages circle through each ring to check its consistency. If a node doesn’t receive such a message in one of its rings for a given amount of time, it checks for link breakage or a partition in the network. If the SAP detects either of these cases, it initiates an appropriate algorithm to repair the ring. Similarly Lanes and Kozat and Tassiulas’s SDP use different algorithms to maintain the lane structure and the dominating set feature of the virtual backbone. Tyan and Mahmoud’s SDP has two methods for supporting mobility. The first approach is that when a gateway node moves to another cell, it broadcasts the service information to nodes in its previous cell. These nodes elect another gateway node. The second method specifies a time-to-live parameter, which is the clock time after which a service must refresh its advertisement.

**SDP Architectural Views**

Broadly speaking, architecture specifies a structure’s layout and how its major components connect. An architectural view would thus be a specific perspective for looking at the SDPs.

**Directory-based vs. Directory-less**

In multihop manet SDPs, the architecture can be directory-based or directory-less. In a directory-based architecture, the user chooses certain nodes in a manet to be directory nodes. The registration and discovery mechanisms are specially tailored to access these nodes. For example, the Lanes, Splendor, Kozat and Tassiulas, and Tyan and Mahmoud SDPs are all directory-based.

Directory-less architecture has no directory nodes, as with SDPs by Liang Cheng and Ivan Marsic, Varshavsky, Reid, and de Lara, Lenders, May, and Plattner, as well as group-based distributed service discovery (GSD), Allia, Konark, Service Rings, and distributed service discovery (DSD). The registration and discovery mechanisms approach all of the network’s nodes or a subset. Manets often use the directory-less approach because it’s difficult to host a directory in mobile nodes that have scarce resources, such as memory and power, and whose availability isn’t guaranteed. A directory node should have vast resources and remain available in the network for frequent contact with other nodes.

**Overlay-based vs. Overlay-less**

The overlay network in multihop manet SDPs has the advantage of being able to control the multicast of service query or advertisement messages. This controlled multicast restricts and greatly reduces the network traffic, as in SDPs by Kozat and Tassiulas, Tyan and Mahmoud, Allia, Service Rings, and Lanes.

In mobile environments, forming and maintaining overlay networks requires resource-constrained nodes to
be vigilant about topological changes and to run algorithms, both resource-consuming tasks. Overlay-less SDPs include those by Cheng and Marsic,\textsuperscript{4} Varshavsky, Reid, and de Lara,\textsuperscript{12} Lenders, May, and Plattner Splendor,\textsuperscript{11} GSD,\textsuperscript{5} Konark,\textsuperscript{7} and DSD.\textsuperscript{15}

**Directory and Overlay**

Rather than looking at SDPs in terms of directory or overlay, let’s look at a SDP in terms of directory and overlay. SDP architectures then fall into the following four categories:

- directory-based with overlay support,\textsuperscript{8–10,13}
- directory-based without overlay support,\textsuperscript{11}
- directory-less with overlay support,\textsuperscript{6}
- directory-less without overlay support.\textsuperscript{4,5,7,12,14,15}

Each of these categories suits different environmental conditions.

**Protocol Suitability**

Different dimensions help us understand SPD behavior—for example, network size, node mobility, and SPD behavior in a network with multiple service instances. Figure 3 shows SPD suitability for different network sizes and mobility conditions. Network size and mobility magnitudes are as follows:

- Small network: up to 10 nodes.
- Medium network: 10 to 100 nodes.
- Large network: greater than 100 nodes.
- Low mobility: up to 5 km per hour (such as in a building).
- Medium mobility: 5 to 50 km per hour (such as a traffic scenario in a city).
- High mobility: greater than 50 km per hour (such as highway traffic speeds).

The shaded region in Figure 3 shows the region in which all SDPs will work well. Unshaded regions suit only particular types of protocols.

**Small networks and high mobility.** Protocols that are directory-less and don’t form overlay networks are suitable for small networks with high mobility. Such protocols use flooding for registration and discovery mechanisms. Flooding has the negative side of generating large overhead packets, making scalability an issue. Thus, SDPs that use flooding aren’t suitable for large networks. Conversely, such protocols don’t form overlays or have directory nodes, so they don’t have the related overhead of maintaining the overlay structure and selecting directory nodes. These protocols perform well under high mobility—for example, the SDP by Cheng and Marsic\textsuperscript{4} shown in Figure 3.

**Medium networks and medium mobility.** Protocols that use limited flooding are well suited for medium size networks with medium speeds of nodes. Examples include flooding limited by time-to-live constraints, multicasting to a group of nodes as in GSD,\textsuperscript{5} or flooding on top of an overlay network as in SDP by Kozat and Tassiulas to form VBNs.\textsuperscript{10} Limited floocasting improves mobility handling, and using overlay networks improves scalability. Figure 3 shows that SDPs that are directory-based without overlay support or directory-less with overlay support are well suited for medium networks and medium mobility.

**Large networks and low mobility.** Overlay networks provide an efficient routing mechanism for searching. Low mobility has no overlay maintenance issues, so SDPs that form overlay networks, such as Service Rings\textsuperscript{8} and Lanes,\textsuperscript{9} perform well in large networks. However, overlay maintenance becomes an issue under high mobility, and such protocols are unsuitable in high mobility scenarios. Figure 3 shows that the SDPs that are directory-based with overlay support are exclusively suited for this region.

**Multiple service instances.** If a network has many instances of the same service, SDPs that integrate search and selection with routing protocols perform better than those with separate search,
selection, and routing mechanisms. In such cases, the search success increases with the increase of service instances. Other advantages include reduced message overhead and less latency in searching the suitable services. Lenders, May, and Plattner’s field theoretic SDP, for example, is well suited for an environment with multiple service instances.14

Service discovery in mobile and pervasive environments is an important and active research field. (Table 1 summarizes our analysis.) Still, many challenges remain to be resolved before SDPs for multi-hop manets can be practical, for example, determining which type of overlay suits manet environments under high mobility.

Often, developers build existing SDPs as standard applications without taking advantage of cross-layering optimizations. These SDPs don’t exploit the wireless medium’s broadcast nature. We must look into new search techniques that exploit broadcast to become aware of network services by simply listening (eavesdropping) to the traffic other nodes generate.

Another ongoing challenge in manet SDPs is limiting network traffic. Typical methods include limited flooding, multicasting, or overlays. A less explored approach is replicating services. We must further explore this approach and tackle issues such as service consistency, as well as number and placement of replications in the network.

Additionally, some peer-to-peer networks use random walk for searching. Little research has gone into this approach for search and discovery in manet SDPs. One critical aspect is reducing the number of hops, which random walk addresses with biasing mechanisms. Still an open question is finding an efficient biasing mechanism for wireless mobile environments.

Finally, different researchers often define their own metrics to measure performance of their proposed SDPs. It becomes impossible to realistically compare SDPs. We need standard performance metrics in this field.17

### Table 1: Service Discovery Protocol Features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Cheng and Marsic SDP(^4)</th>
<th>GSD SDP(^5)</th>
<th>Allia SDP(^6)</th>
<th>Konark SDP(^7)</th>
<th>Service Rings(^8)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service discovery architecture</td>
<td>Directory-less without overlay support</td>
<td>Directory-less without overlay support</td>
<td>Directory-less without overlay support</td>
<td>Directory-less without overlay support</td>
<td>Directory-based with overlay support</td>
</tr>
<tr>
<td>Management of service information</td>
<td>Stores information of local services and multicast service advertisements</td>
<td>Multicast service advertisements to (n) hops</td>
<td>Broadcast service advertisements to 1 hop (immediate neighbors)</td>
<td>Multicast service advertisements</td>
<td>Service information stored in service rings’ service access point nodes</td>
</tr>
<tr>
<td>Search methods</td>
<td>Multicast query request to a multicast group</td>
<td>Query request selectively forwarded based on semantic information</td>
<td>Query request first multicast to alliance member, then to other alliances</td>
<td>Multicast query request</td>
<td>Service access points</td>
</tr>
<tr>
<td>Service selection methods</td>
<td>Not specified</td>
<td>Not specified</td>
<td>Not specified</td>
<td>Not specified</td>
<td>Not specified</td>
</tr>
<tr>
<td>Mobility support methods</td>
<td>Multicast service advertisements</td>
<td>Adjust service advertisement diameter and rate</td>
<td>Adjust advertisement rate and alliance diameter</td>
<td>Specify lifespan of service advertisements and periodical advertisements</td>
<td>Repair service rings damaged due to mobility</td>
</tr>
<tr>
<td>Service description methods</td>
<td>Not specified</td>
<td>DARPA Agent Markup Language (DAML)</td>
<td>Independent of description language</td>
<td>Extensible Markup Language (XML)</td>
<td>Any language that describes and summarizes functions</td>
</tr>
</tbody>
</table>

---

**Note:**
- CHENG AND MARSCI SDP: Directory-less without overlay support
- GSD SDP: Directory-less without overlay support
- ALLIA SDP: Directory-less without overlay support
- KONARK SDP: Directory-less without overlay support
- SERVICE RINGS: Directory-based with overlay support
- SOFTWARE: Multicast service advertisements to \(n\) hops
- REQUEST: Broadcast service advertisements to 1 hop (immediate neighbors)
- CONTENT: Multicast service advertisements
- MULTICAST: Service information stored in service rings’ service access point nodes
- METHODS: Multicast query request
- SELECTION: Not specified
- MOBILITY: Multicast service advertisements
- DESCRIPTION: Not specified
- LANGUAGE: DARPA Agent Markup Language (DAML)
- INDEPENDENT: Independent of description language
- XML: Extensible Markup Language (XML)

---

**Table 1:**

<table>
<thead>
<tr>
<th>Feature</th>
<th>Cheng and Marsic SDP(^4)</th>
<th>GSD SDP(^5)</th>
<th>Allia SDP(^6)</th>
<th>Konark SDP(^7)</th>
<th>Service Rings(^8)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service discovery architecture</td>
<td>Directory-less without overlay support</td>
<td>Directory-less without overlay support</td>
<td>Directory-less without overlay support</td>
<td>Directory-less without overlay support</td>
<td>Directory-based with overlay support</td>
</tr>
<tr>
<td>Management of service information</td>
<td>Stores information of local services and multicast service advertisements</td>
<td>Multicast service advertisements to (n) hops</td>
<td>Broadcast service advertisements to 1 hop (immediate neighbors)</td>
<td>Multicast service advertisements</td>
<td>Service information stored in service rings’ service access point nodes</td>
</tr>
<tr>
<td>Search methods</td>
<td>Multicast query request to a multicast group</td>
<td>Query request selectively forwarded based on semantic information</td>
<td>Query request first multicast to alliance member, then to other alliances</td>
<td>Multicast query request</td>
<td>Service access points</td>
</tr>
<tr>
<td>Service selection methods</td>
<td>Not specified</td>
<td>Not specified</td>
<td>Not specified</td>
<td>Not specified</td>
<td>Not specified</td>
</tr>
<tr>
<td>Mobility support methods</td>
<td>Multicast service advertisements</td>
<td>Adjust service advertisement diameter and rate</td>
<td>Adjust advertisement rate and alliance diameter</td>
<td>Specify lifespan of service advertisements and periodical advertisements</td>
<td>Repair service rings damaged due to mobility</td>
</tr>
<tr>
<td>Service description methods</td>
<td>Not specified</td>
<td>DARPA Agent Markup Language (DAML)</td>
<td>Independent of description language</td>
<td>Extensible Markup Language (XML)</td>
<td>Any language that describes and summarizes functions</td>
</tr>
<tr>
<td>Service Discovery Protocols</td>
<td>Features</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------------------------</td>
<td>----------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lanes SDP</td>
<td>Directory-based with overlay support</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Splendor SDP</td>
<td>Directory-based with overlay support</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Varshavsky, Reid, and de Lara SDP</td>
<td>Directory-based without overlay support</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tyran and Mahmoud SDP</td>
<td>Directory-less without overlay support</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lenders, May, and Plattner SDP</td>
<td>Directory-based without overlay support</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Distributed service discovery SDP</td>
<td>Directory-less without overlay support</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Directory-based with overlay support**: Stores information in a directory that is replicated in virtual backbone nodes. Gateway node keeps track of services in a group. Multicast service advertisements to a hop with selective forwarding.
- **Directory-based without overlay support**: Used in service-oriented architectures. Gateway node keeps track of services in a group. Multicast service advertisements to a hop with selective forwarding.
- **Directory-less with overlay support**: Anycast query request to nodes flooded only to virtual backbone nodes. Query routed to neighbors with higher potential. Query selectively forwarded based on ontology descriptions.
- **Not specified**: Selection done at client side (no further detail given). Selection based on lowest hop count. Distributed selection based on network distance and capacity of service. Not specified.
- **Maintenance of virtual backbones’ dominating set feature**: Specify lifespan of service advertisements and periodical advertisements. Periodical advertisements. Adjust advertisement diameter and rate.

### ACKNOWLEDGMENTS

The Italian Ministry of Education, University, and Research (under the IS-Manet project) and the Resilience for Survivability in IST Network of Excellence program (European Commission contract no. 026764) supported this project.

### REFERENCES

10. U. Kozat and L. Tassiulas, “Network...


For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/cslid.