CSE261: Computer networks
Course project: OSPF(bare bone)
Phase 2 (final) : OSPF communication

GALALA UNIVERSITY
Powered by Arizona State University

Important note: Plagiarism (copy and paste), from colleagues or from other sources (e.g online sources) will be punished, you risk losing the entire phase marks. You are responsible for developing your own submission.

Phase Two objectives:
To simulate the OSPF protocol and get familiar with different components of the protocol.

Phase Two deliverables:

1. Python program
2. Phase Two report that includes:
    a. Section one: A brief introduction about the OSPF protocol, including some information about LSA types 1,2,3 and LSU and LSBD.
    b. Section Two: result section that includes one test case for the code and its results (actual figures of the code output) as well as a brief comment on the results. The code aboutput should be the LSBDs of different routers and their forwarding tables.
    c. Appendix section: your python code.

Phase two marks:

- 10 marks out of the total 20 marks dedicated to the course project.

Phase Two deadline:

- Please refer to the canvas page.

CSE261: Computer networks
Course project: OSPF(bare bone)
Phase 2 (final) : OSPF communication

GALALA UNIVERSITY
Powered by Arizona State University

Phase two in detail:
Phase two will create and use two main entity to emulate the OSPF process:

1. OSPF LSA: this is the message sent by an OSPF router to give other routers link information as well as other important network information. You should use the following LSA form (which is not exact but similar to the real one.)

(Table 1 modified LSA format)

Link State ID	The link ID, in this project we use connected routers to ID the link, for example we used link (1,2) to ID a link between router 1, and router 2. In reality it is something like 10.1.1.1, each router then has to investigate different LSAs to know which routers are attached to 10.1.1.1. In this project we skip the search part to keep it simple
Sequence Number	Each LSA has a sequence number to differentiate between them. In this project we use a sequence number made of four numbers ($N_1N_2N_3N_4$). N_1N_2 is the router number and the N3N4 is the message number. For example the first message sent by the 12th router has a seq. Number (1201)
Advertising Router	The name of the original router that sent the link state information (e.g router 12).
Link Cost	A metric used by OSPF to rank the links based on bandwidth, delay (i.e. weight in this project)
Timestamp	The time at which the message was created.

CSE261: Computer networks
Course project: OSPF(bare bone)
Phase 2 (final) : OSPF communication

GALALA UNIVERSITY
Powered by Arizona State University

---

2. OSPF LDSB

All LSAs received by a certain router are then stored in the LDSB database that should look like this

(Table 2 modified LSDB format)

Sequence number	Advertising Router	Link State ID	Timestamp	Link Cost

Example for a real world LSDB

```
Router5# show ip ospf database

 OSPF Router with ID (5.5.5.5)

        Router Link States (Area 0)

Link ID     ADV Router     Age    Seq#       Checksum Links
1.1.1.1     1.1.1.1        783    0x80000005  0x00FAEB 7
2.2.2.2     2.2.2.2        748    0x80000005  0x0031A9 9
3.3.3.3     3.3.3.3        727    0x80000004  0x00701F 7
4.4.4.4     4.4.4.4        746    0x80000003  0x00240B 8
5.5.5.5     5.5.5.5        739    0x80000003  0x00C036 2

        Net Link States (Area 0)

Link ID     ADV Router     Age    Seq#       Checksum
10.0.1.1    1.1.1.1        783    0x80000004  0x0039D1
10.0.2.2    2.2.2.2        748    0x80000004  0x00F50F
10.0.3.3    3.3.3.3        727    0x80000003  0x00A45B
```

CSE261: Computer networks
Course project: OSPF(bare bone)
Phase 2 (final) : OSPF communication

GALALA UNIVERSITY
Powered by Arizona State University

Phase two functions in detail:
Refer to the figure below

These are the main functions that each router's program should perform. If you have five routers, you would need to run five programs simultaneously on different machines or on the same machine.

Communication part :
(this part should run as first as the program starts)

1. Sending link state information (link weight and link connected routers (its position in the graph)) from routers that are directly connected to other routers that are not connected to this link. This message is called (LSA). For our simulation, simply store the link weights in a separate file for each router, i.e. each router should read the weights of the links connected to it from a file designated to it.
2. The router should then prepare an LSA message to send the information mentioned above in the form described in the above tables (table 1). Each router should send its LSA to other routers connected directly to it. You can keep this information also in the same router file for the router to "know" which routers are connected directly to it. These LSA messages should be sent over a UDP connection to the intended routers. If you are running the simulations of all routers on the same machine, assign different port numbers to different routers.
3. Each router then stores all LSA messages received in a database (OSPF LSDB).
4. Routers should forward the LSA messages received from one router to the other routers. To avoid looping make sure not to forward an LSA twice and do not send it back to the relaying router. You can achieve that using these two measure:
    - Calling the functions of LSBD building and LSA forwarding together.
    - When receiving a new LSA, check its sequence number against the LSDB. If it is already stored, then discard the LSA and not forward it.

Building Forwarding table part (phase one):
(this should run after some delay (let us put it at 10 secs) to give time for the LSDB to be built first)

1. The program should read the link information from the LSBD and pass it to phase one code to build the forwarding table.

CSE261: Computer networks

Course project: OSPF(bare bone)

Phase 2 (final) : OSPF communication

GALALA UNIVERSITY
Powered by Arizona State University