Distributed Operating System Principles

Programming Assignment 2

Chord: P2P System and Simulation

Satya Aakash Obellaneni

Guna Teja Athota

Kondreddy Rohith Sai Reddy

Sushmanth Karnam

1. Problem Description

Simulate a Chord network with multiple nodes in the Akka.NET framework using F#. Make a Chord network, then implement node joining, stabilization, and efficient routing, as well as calculating the average hops for message delivery, the purpose is to explain and demonstrate Chord's distributed hash table capabilities In a simulated setting

2. Chord Protocol

Chord is a peer-to-peer distributed system protocol in which we create a ring structure with all sparsely distributed nodes and each node has a finger table with key-value pairs that have information on the nodes with information, reducing the search time for a key in node from O(n) to O(log n).

a) Generate a chord ring: We arrange all of the nodes into a ring. We update the ring whenever a node enters or exits the distributed system.
b) Finger Table: Each node has a finger table with m entries (the number of bits in the hash key). The first element in the table represents the node's immediate successor; the remaining entries are nodes that are a logarithmic distance away from the current node. When a node enters or exits the distributed system, the finger table is updated.
c) When a node enters or departs the distributed system, a successor points to the next node in the ring and a predecessor points to the prior node. When a node arrives, we update existing nodes' predecessor, successor, and finger tables. The new node obtains the necessary keys from its predecessor.
d) Look for a node that includes the following key: We use finger tables to locate the node that has the key. Based on the values in the finger table, we go to the next node. This implementation computes the average number of hops.

3. Functionalities Implemented

In this Chord network simulation following are working

a) Starting with the initial node, nodes join the network successively via "JoinChord" signals.
b) Stabilization maintains the ring structure and handles node changes.
c) Finger tables optimize routing and are updated on a regular basis.
d) Messages with defined hops are efficiently routed between nodes.
e) The code computes and displays the average message delivery hops.

4. Running the Code

To run the code, we must give 2 inputs:

Command to implement the code: dotnet fsi chord.fsx <no_of_Nodes><no_of_Requests>

no_of_Nodes: numbr of nodes for the chord ring

no_of_Requests: number of keys each actor searches for

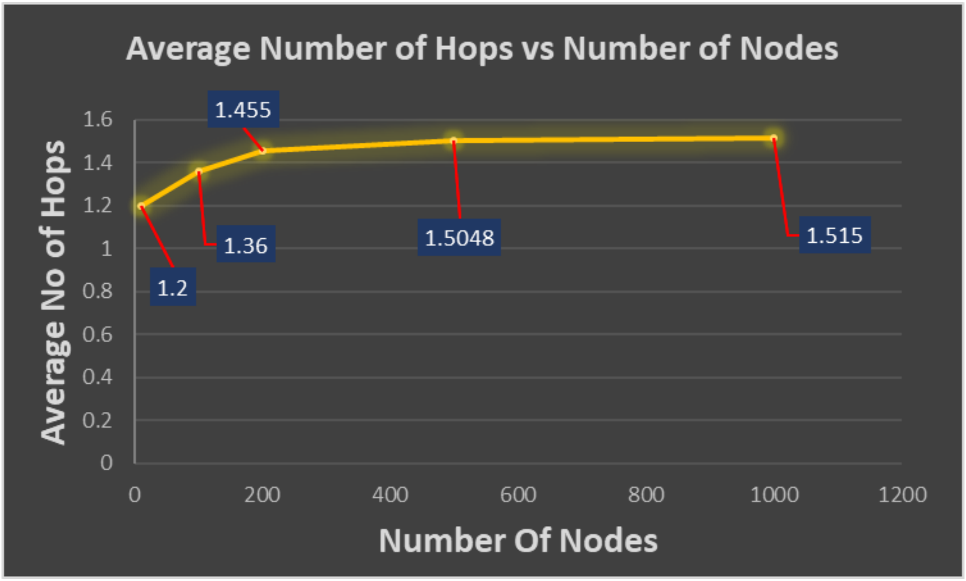
5. Assumptions

- For simulation reasons, it is assumed that the Chord network is built in a local Akka.NET context.
- The node IDs are produced at random using a hash of a random string, guaranteeing that they are unique.
- The network is joined sequentially, with the first node joining itself to establish the network.
- Nodes update their finger tables and execute network stabilization on a regular basis.
- To discover successors and perform message hops, a basic routing method is utilized.
- Here the input number of nodes should be greater than 1.

6. Table

Nodes	No. of Requests	Total Hops	Avg No of Hops
10	5	60	1.2
100	5	680	1.36
200	5	1455	1.455
500	5	3762	1.5048
1000	5	7575	1.515
5000	5	36880	1.4752
10000	5	73460	1.4692
50000	5	364630	1.45852
100000	3	443870	1.47956

7. Graph



8. Largest Network Managed

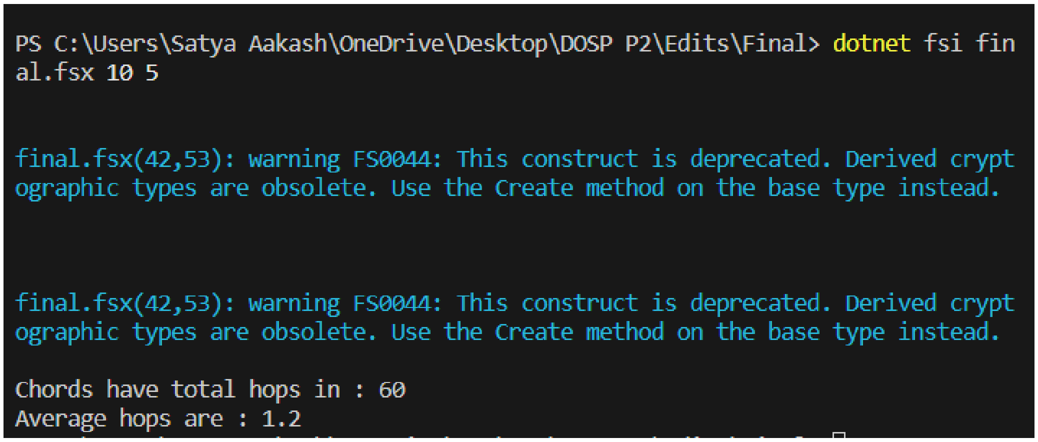
Tried to implement with 100000 nodes and with 3 requests and we got the output.

9. Output

Case 1:

no_of_Nodes: 10

no_of_Requests: 5

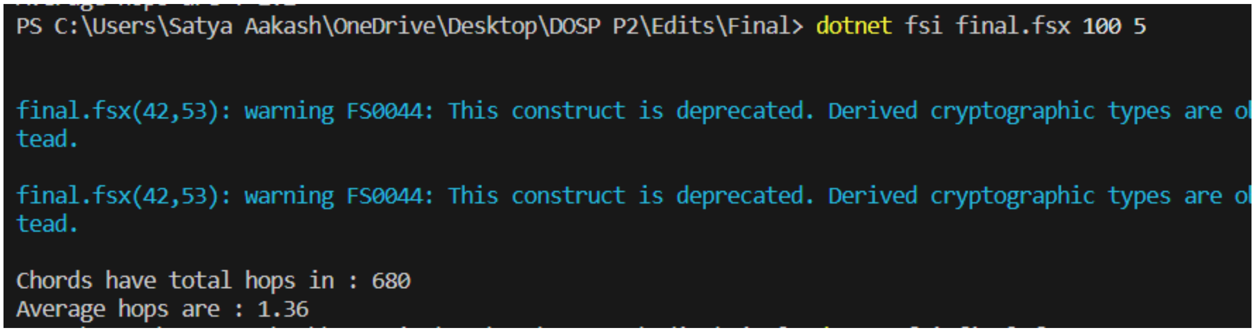


```
PS C:\Users\Satya Aakash\OneDrive\Desktop\DOSP P2\Edits\Final> dotnet fsi fin
al.fsx 10 5


final.fsx(42,53): warning FS0044: This construct is deprecated. Derived crypt
ographic types are obsolete. Use the Create method on the base type instead.



final.fsx(42,53): warning FS0044: This construct is deprecated. Derived crypt
ographic types are obsolete. Use the Create method on the base type instead.

Chords have total hops in : 60
Average hops are : 1.2
```

Case 2:

no_of_Nodes: 100

no_of_Requests: 5

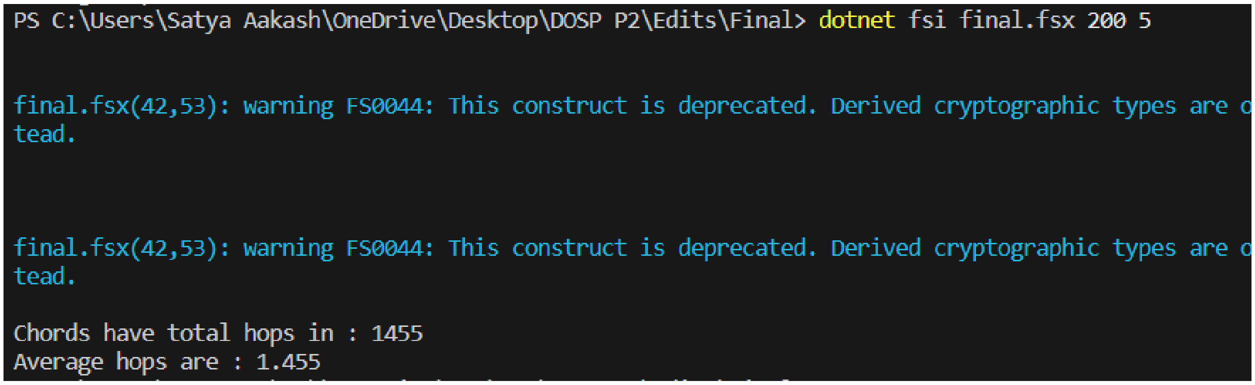


```
PS C:\Users\Satya Aakash\OneDrive\Desktop\DOSP P2\Edits\Final> dotnet fsi final.fsx 100 5


final.fsx(42,53): warning FS0044: This construct is deprecated. Derived cryptographic types are ol
tead.

final.fsx(42,53): warning FS0044: This construct is deprecated. Derived cryptographic types are ol
tead.

Chords have total hops in : 680
Average hops are : 1.36
```

Case 3:

no_of_Nodes: 200

no_of_Requests: 5

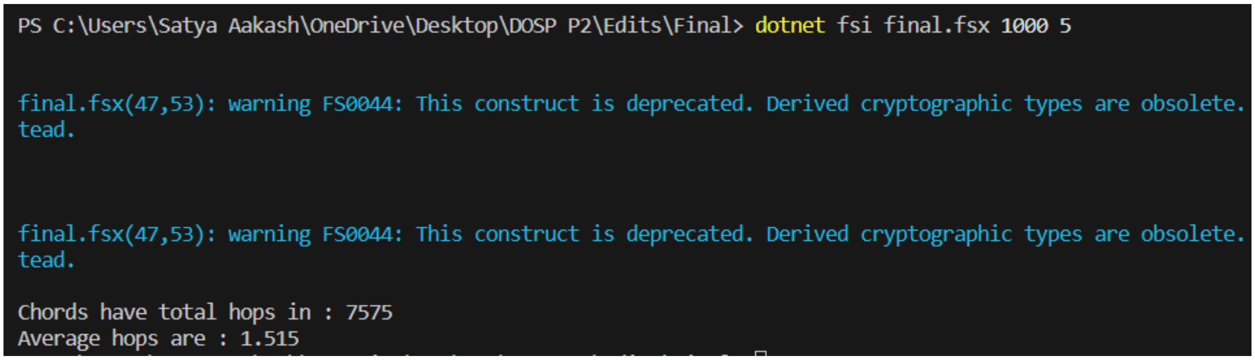


```
PS C:\Users\Satya Aakash\OneDrive\Desktop\DOSP P2\Edits\Final> dotnet fsi final.fsx 200 5


final.fsx(42,53): warning FS0044: This construct is deprecated. Derived cryptographic types are o
tead.



final.fsx(42,53): warning FS0044: This construct is deprecated. Derived cryptographic types are o
tead.

Chords have total hops in : 1455
Average hops are : 1.455
```

Case 4:

no_of_Nodes: 1000

no_of_Requests: 5

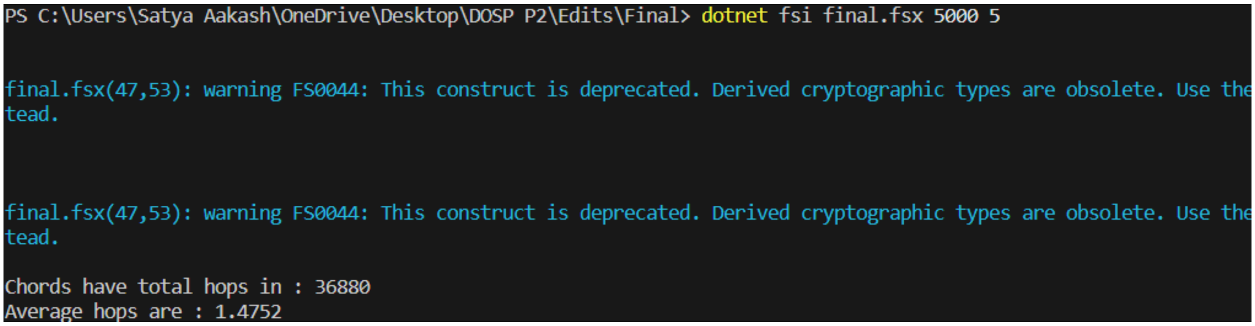


```
PS C:\Users\Satya Aakash\OneDrive\Desktop\DOSP P2\Edits\Final> dotnet fsi final.fsx 1000 5


final.fsx(47,53): warning FS0044: This construct is deprecated. Derived cryptographic types are obsolete.
tead.



final.fsx(47,53): warning FS0044: This construct is deprecated. Derived cryptographic types are obsolete.
tead.

Chords have total hops in : 7575
Average hops are : 1.515
```

Case 5:

no_of_Nodes: 5000

no_of_Requests: 5

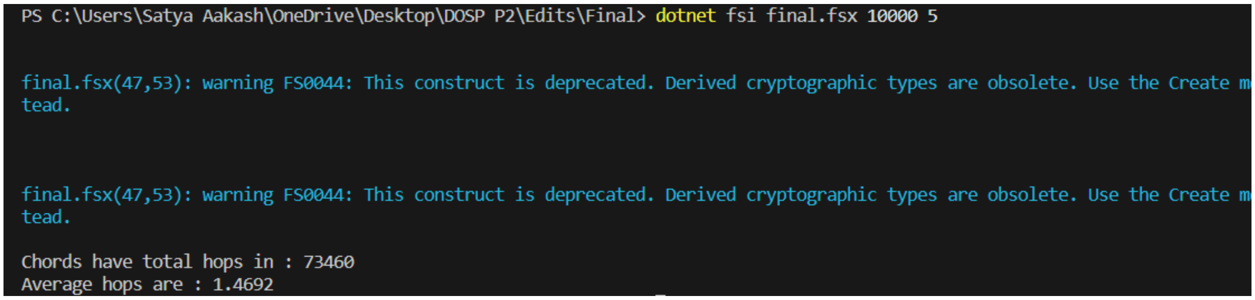


```
PS C:\Users\Satya Aakash\OneDrive\Desktop\DOSP P2\Edits\Final> dotnet fsi final.fsx 5000 5


final.fsx(47,53): warning FS0044: This construct is deprecated. Derived cryptographic types are obsolete. Use the
tead.



final.fsx(47,53): warning FS0044: This construct is deprecated. Derived cryptographic types are obsolete. Use the
tead.

Chords have total hops in : 36880
Average hops are : 1.4752
```

Case 6:

no_of_Nodes: 10000

no_of_Requests: 5

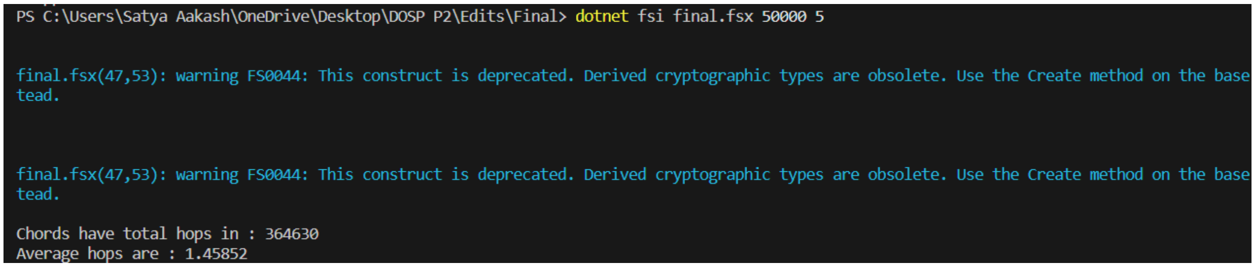


```
PS C:\Users\Satya Aakash\OneDrive\Desktop\DOSP P2\Edits\Final> dotnet fsi final.fsx 10000 5


final.fsx(47,53): warning FS0044: This construct is deprecated. Derived cryptographic types are obsolete. Use the Create m
tead.


final.fsx(47,53): warning FS0044: This construct is deprecated. Derived cryptographic types are obsolete. Use the Create m
tead.

Chords have total hops in : 73460
Average hops are : 1.4692
```

Case 7:

no_of_Nodes: 50000

no_of_Requests: 5

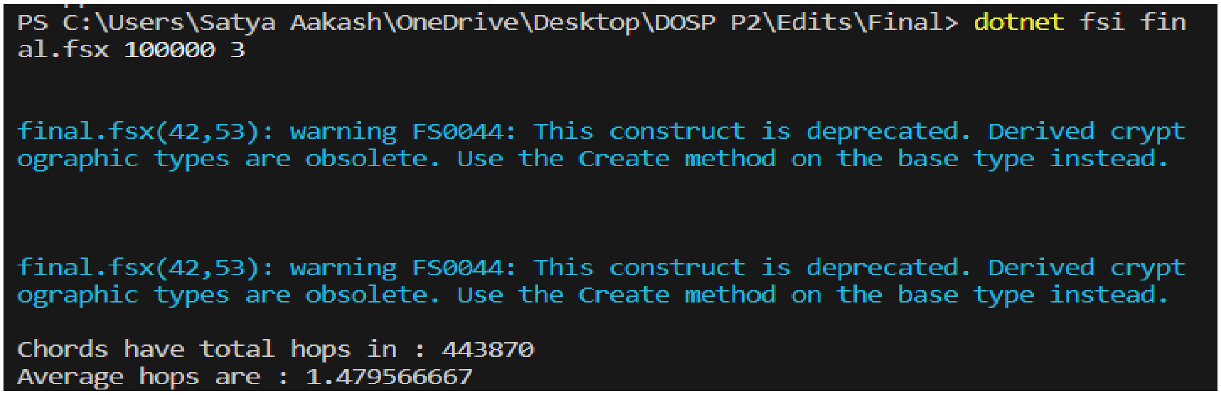


```
PS C:\Users\Satya Aakash\OneDrive\Desktop\DOSP P2\Edits\Final> dotnet fsi final.fsx 50000 5


final.fsx(47,53): warning FS0044: This construct is deprecated. Derived cryptographic types are obsolete. Use the Create method on the base
tead.


final.fsx(47,53): warning FS0044: This construct is deprecated. Derived cryptographic types are obsolete. Use the Create method on the base
tead.

Chords have total hops in : 364630
Average hops are : 1.45852
```

Case 8:

no_of_Nodes: 100000

no_of_Requests: 3



```
PS C:\Users\Satya Aakash\OneDrive\Desktop\DOSP P2\Edits\Final> dotnet fsi fin
al.fsx 100000 3


final.fsx(42,53): warning FS0044: This construct is deprecated. Derived crypt
ographic types are obsolete. Use the Create method on the base type instead.


final.fsx(42,53): warning FS0044: This construct is deprecated. Derived crypt
ographic types are obsolete. Use the Create method on the base type instead.

Chords have total hops in : 443870
Average hops are : 1.479566667
```