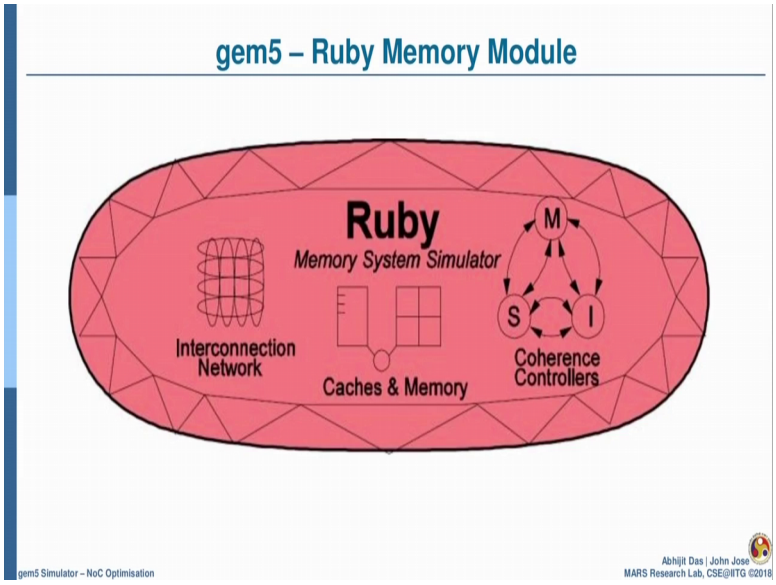
Multi-Core Computer Architectures-Storage and Interconnects
Prof. Abhijit Das
Department of Computer Science and Engineering
Indian Institute of Technology, Guwahati

Lecture - 23
gem5 Simulator - NoC Optimisation

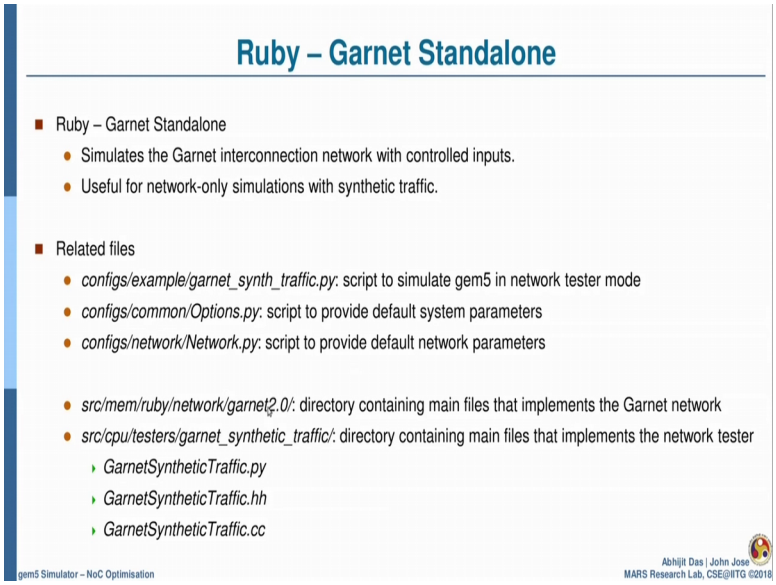
Welcome everyone. I am Abhijit Das. And today, I will give you the pointers on how to solve assignment-6 in our NPTEL course Multi-Core Computer Architectures- Storage and Interconnects. Assignment-6 is a simulation based assignment to be carried out on gem5 simulator. We have already posted a video on how to install and get started with gem5 simulator. Assignment-6 is about network on chip or NoC Optimization, where you need to change certain NoC related parameters and assess the performance.

(Refer Slide Time: 01:08)



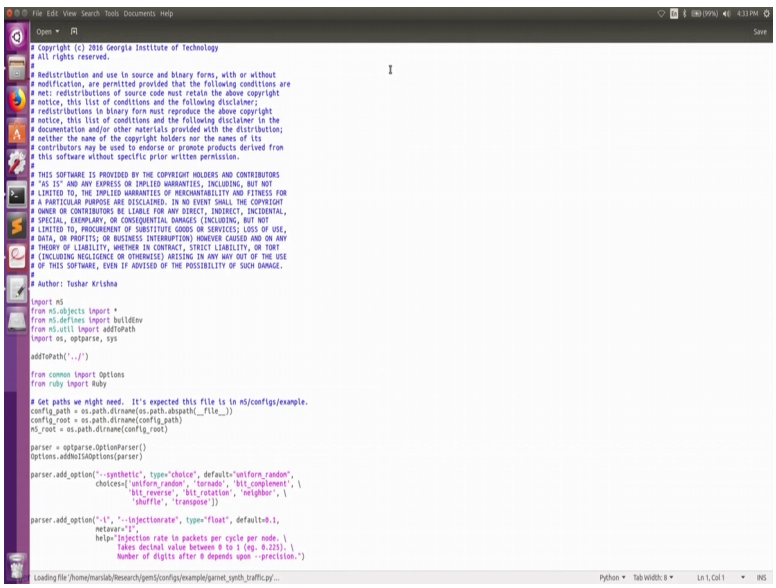
Gem5 has a module called ruby, which takes care of the entire memory system. Ruby in turn contains three different modules called interconnection network, caches and memory, and coherence controllers. For solving assignment-6, we will mainly focus on this module.

(Refer Slide Time: 01:30)



Garnet standalone module in ruby simulates the Garnet interconnection network with controlled inputs. And it is mainly useful for network only simulations with synthetic traffics. Meaning there is no actual processor and memory interaction, and only the network is artificially simulated. These are the related files, where Garnet synthetic traffic dot py is the script to simulate a gem5 in network tester mode. Options dot py is the script to provide default system parameters. And network dot py is the script to provide default network parameters. Let me show you, where these files are in the gem5 directory.

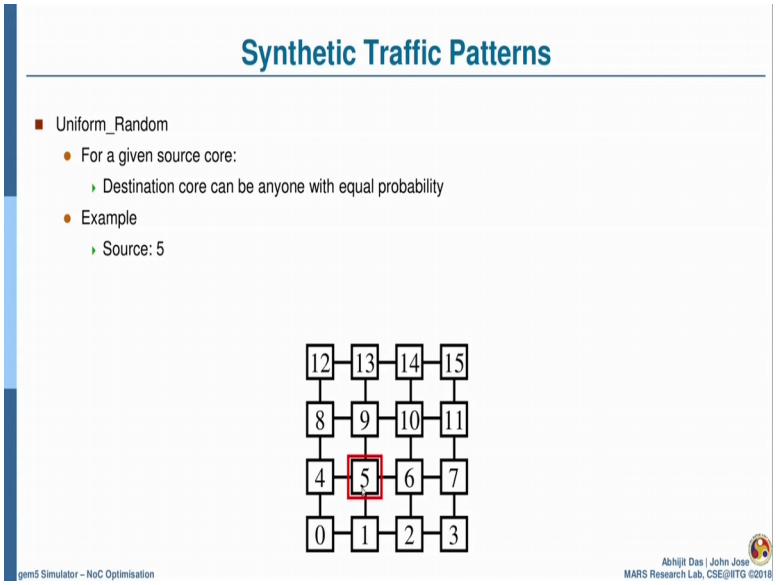
(Refer Slide Time: 02:26)



I am inside gem5 directory. And the very first configuration script is inside example and Garnet synthetic traffic dot py. Here we can see a number of different parameters with their default values. Going back to the directory, the next file is in commons options dot py, where again you can see different parameters like memory size l 1, data cache size l 1, instruction cache size with their default values.

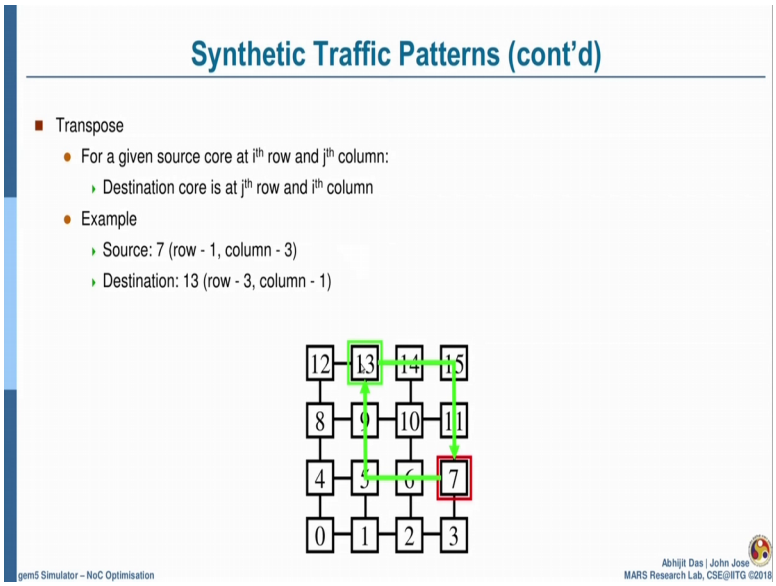
The next configuration script is in network called network dot py, which has network specific parameters with their default values. Going back to the slides, ruby module has a directory called garnet 2.0, which contains the main files that implements the garnet network. And CPU testers contains another directory called garnet synthetic traffic, which specifically contains the files that implements the network tester.

(Refer Slide Time: 03:35)



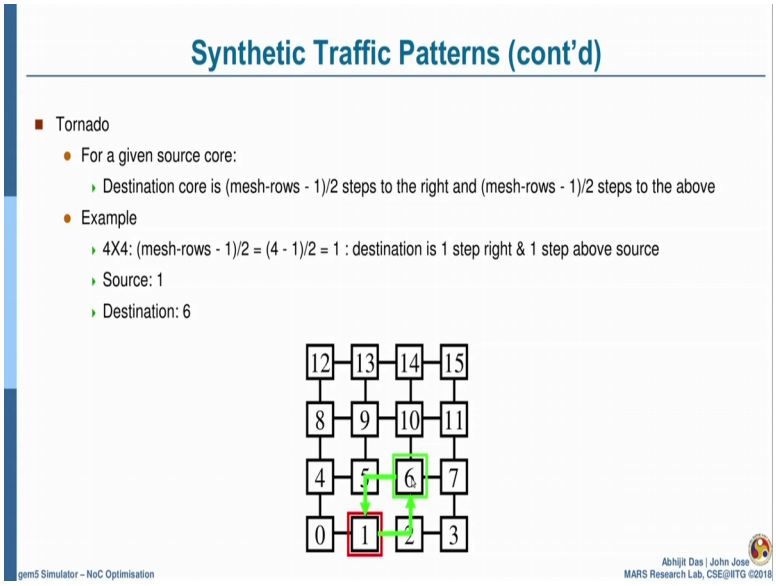
Since, there are no actual processor and memory interaction in network tester mode, we make use of some synthetic traffic patterns. And one of them is called uniform random, where for a given source core, the destination core can be anyone with equal probability. To take an example, suppose in this 4 cross 4 network, the source core is 5. So, according to uniform random traffic pattern, the destination for this core can be anyone other than itself.

(Refer Slide Time: 04:15)



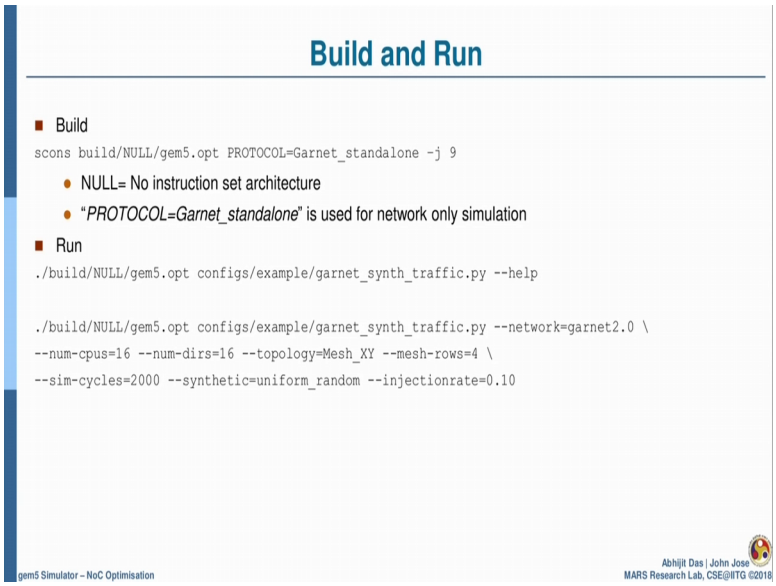
The next synthetic traffic pattern is called transpose, where for a given source core at ith row and jth column, the destination core is at the transpose that is at jth row and ith column. For example, if the source core is 7, we count the row as row 0, row 1, row 2, row 3, and columns as column 0, column 1, column 2, and column 3, source 7 is in row-1, and column-3. So, according to transpose traffic pattern, the destination is 13, because its row and column is just the transpose of the source core 13 is in row-3, and column-1.

(Refer Slide Time: 05:09)



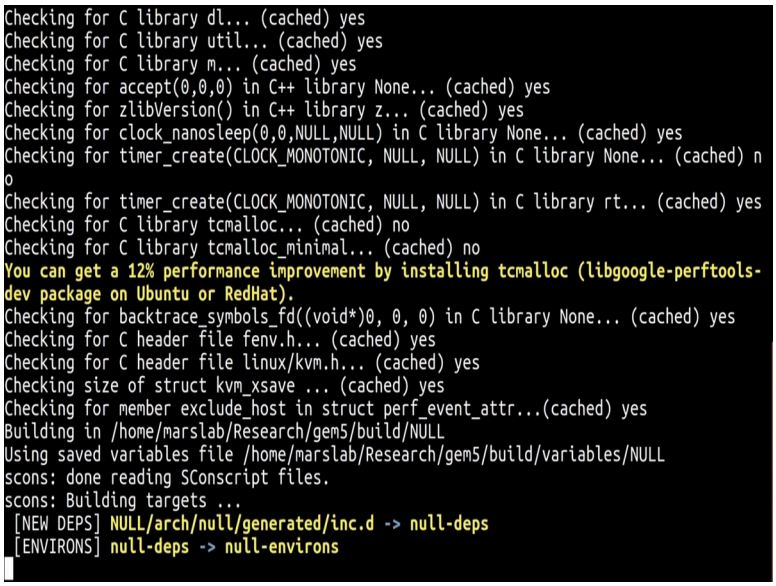
The next synthetic traffic pattern is tornado, where for a given source the destination core is at mesh-rows minus 1 by 2 steps to the right and mesh-rows minus 1 by 2 steps to the above. To understand with an example for a 4 cross 4 network here, this formula will give me a value 1, which means that my destination is 1 step right and 1 step above the source. Suppose, my source is 1, so according to this formula my destination is 1 step right and 1 step above. So, 6 is my destination here.

(Refer Slide Time: 05:54)



After the general understanding of Garnet standalone module in ruby, we can use this command to get started with the build. Here NULL is no instruction set architecture, because we are not simulating actual processors and memory. So, we need not specify an actual instruction set architecture. Next is the gem5.opt binary. And next is PROTOCOL is equal to Garnet standalone, which is used for network only simulation. Let me show you, how to do this.

(Refer Slide Time: 06:29)



I have to go to the gem5 directory. And there, I will try to build null architecture with gem5.opt binary, and Garnet standalone PROTOCOL. The number of jobs I specify is 9, because I have 8 CPU's in my system and the build begins. This message is the assurance that my build is successful. Going back to the slides; after the successful build, we can get started with the simulation run using this command. And again we can use something like minus minus help with this configuration script to get the available help options.

Since, I have already shown it in the previous video I am not showing it here. Then I can run a sample simulation, let me show you. I have to run NULL isa with gem5.opt, and the configuration script that I want to specify is garnet synthetic. And I want to specify the network as garnet, I want to specify the number of source, which is also called number of cpus as 16. The number of destinations, which is also called number of directories as 16 and the topology as Mesh XY; I have to specify the number of mesh-rows in my system, and since I am simulating a 16 core system, the number of mesh rows is 4.

Next, I can specify for how many simulation cycles, I want to run my simulation say I want to run for 1000 cycles. And I have to specify the synthetic traffic pattern, let me use uniform random, I can specify injection rate in the network, let me specify it as 0.1. I

have an error, I can see that the error is here, I have not used minus minus before this parameter.

So, let me go and correct it. The run is successful, you can ignore these warnings. And directly go to the stats file in the m5 directory, to get the output values. I am inside m5 directory, and this is the stats file. I can search network related parameters. Here this is the number of flits that the network received, this is the number of flits that the network injected, this is the network latency for the flits, this is the queuing latency for the flits and so on. You have to use these values to calculate different results according to the questions given in assignment-6.

Going back to the terminal; we can change these parameters as per the requirement of your questions in assignment-6. For example, if I want to change the injection rate to 0.2, I can run it again, and it will give me different results. For your assignment-6 some of the parameters has to be changed in the terminal, whereas other parameters has to be changed in the files. Suppose, I want to change these parameter called link with bits to 128, I can change it here, save the file, and run my simulation again.

The simulation is successful. We can go back to the same stats file to get output values, and calculate our results. Going back to the slides, this is the command that I just ran. You can go through the following references for doubts clarification. However, if you have any specific doubts with respect to assignment-6, you can get back to us on the discussion forum in the course portal.

Thank you.