Lab 2 Routing and Selection

I. Purpose

In this Lab, we will let you run some experiments with different routings to observe the properties of NoC on our Access Noxim. Please read the background information and follow the procedure and learn running a set of simulation. Hope you will enjoy using our tool!

II. Introduction

A. Noxim Explorer

In NoC simulation, we need to set different parameters to explore the huge design space. Simulation time is quite long and need a set of experiment. Thus we can use Noxim explorer to explore each configuration of the design space generated by configuration file and exports results in Matlab format.

III. Procedure

A. Noxim Explorer

1. Switch the directory to /other, use vim to open the configuration file "sim.cfg"

```
% cd <INSTALL DIRECTORY>/other
% vim sim.cfg
```

2. Modify the topology first, set topology as an 8 by 8, 4-layer NoC.

```
[topology]
   8x8x4
[/topology]
```

3. Leave the routing and selection tag untouched. You can modify here, while you need

change the routing or selection function.

```
[routing]
    xyz
[/routing]
[sel]
    random
[/sel]
```

4. Modify the exploration interval of PIR(packet injection rate). While the first parameter means start point, second represent end point, third is the step size and the last is the distribution pattern. We leave distribution pattern untouched and modify the interval to explorer the design space.

```
[pir]
    0.001 0.050 0.005 poisson
[/pir]
```

5. There are some default settings, you can change the simulation time, warm-up time, packet size and buffer size here. If you want add some self-definition parameters, please add your parameters here. Leave the aggregation tag untouched, pir is the main design space index we use.

```
[default]
    -sim 100000
    -warmup 10000
    -size 8 8
    -buffer 4
[/default]
[aggregation]
    pir
[/aggregation]
```

6. In default case, the path of noxim is "../bin/Noxim". If execution path changed, please modify here.

```
[explorer]
    simulator ../bin/noxim
    repetitions 1
[/explorer]
```

7. Now, save the configuration file and then compile the Noxim explorer to explore the design space of NoC! After simulation ends, use vim to open the log file.
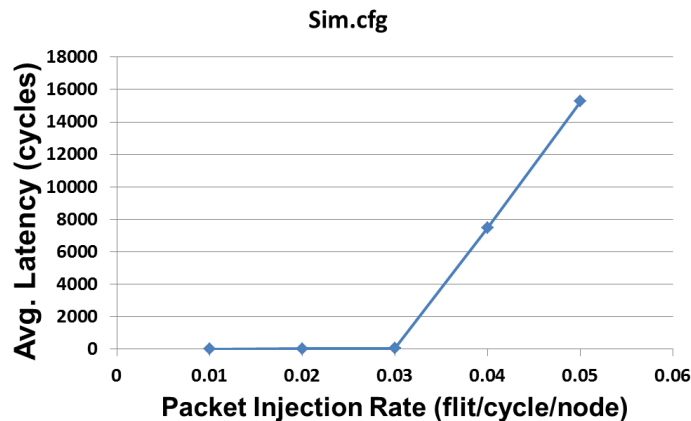
```
[shift] + [;], wq + [enter]
%make
%noxim_explorer sim.cfg
```

```
% vim routing_xyz_sel_random_topology_8x8x4_.m
```

8. If you have Matlab, you can execute the function directly.

```
>> routing_xyz_sel_random_topology_8x8x4_('r-x')
```

9. Or you can use other software to plot the PIR v.s. Avg. Latency figure.

10. We usually focus on the interval that avg. latency is lower than 100. Now, please find the interval and run a set of simulation at least 20 data points in that interval.

Start point	End point	Step Size
0.001		

IV. Problems

A. Routing function

Under the same traffic pattern, different routing algorithm has different performance. Please run an experiment to show the performance between XYZ("XYZ") and OddEven ("oe_z") routing under "random" and "transpose1" traffic. Then plot the average latency v.s. PIR figure to identify your result.
(Hint: put the "–traffic" tag into sim.cfg file, and use "random" and "transpose1" to switch traffic pattern.)

B. Selection function

While we use adaptive routing algorithm, routing function will give us more than one

direction to route the packet. We need to decide which way to route to get a better performance. You have to design a set of experiment to observe the property of random, buffer-level and NoP selection. Then plot the average latency v.s. PIR figure to identify your result.

(Hint: Change the selection function in "-sel")

C. Deadlock

Try to draw the turn model of North-first in 2D and 3D NoC. Is there any possibility to form a dead-lock? Please explain. If we want to implement the Valiant's Randomized Routing in our NoC, what's the potential problem? How do you fix it?