Design Document : Assignment 1 - Parallel Computing(CS F422)

Name: Aditya Sarma A.S, Mihir Biswal
ID: 2013A7PS079P, 2012B2A7767P.

Implementation has been done using the MPICH2 implementation of the MPI Standard. The Language used is C. Our basic principle is this: There is a source process which generates some original news(numNews) and keeps generating updates to the previously generated original news. Each update / original news has an ID in the range 0-newsId-1 using which we can identify which update is an update to which original news. News is generated until total generated content is equal to TOTAL_NEWS.

Basic Design Adopted:

BNPS:

We have used only a single source for news generation. The number of Editors is 1 as per specification. The number of reporters can be changed as required. The number of news updates after which termination should be done can also be set(TOTAL_NEWS). Mapping has been done as to which news update should go to which reporter. It has been made sure that there is some sharing between between reporters in the mapping so that reporter reporter communication also happens. The source makes this mapping and sends this data initially to all the processes involved. This communication is Blocking in nature.

Source:

The source works in the following way. It has a fixed number of events that it will cover(given by numNews). All the other news that the source generates is an update to one of the above generated numNews . An update / news is generated once every second. In the end, the reporter will send a Termination message to all the processes. In reception of this message the processes will finalize the pending communications and exit.

Editor:

The editor receives a news update from a reporter and it publishes it right away, since the Reporter-Reporter communication makes sure that only the latest news reaches the Editor. Validity check has not been implemented as such. The content inside a news item is random. But an improvisation can be made. If the editor receives an update with timestamp t and the last published news had a timestamp t` such that t' > t then the editor will ignore this news. Editor exits when it gets message when it gets a Termination message from the Source.

Reporter:

The reporter receives news from the Source. It sees whether this newsId is exclusively for itself or it is shared. If exclusive it sends this update to the Editor who will publish it. Else, the reporter will send a request message to all the reporters with which the news is shared to compare if its update is latest. If yes, it will inform others to not send this particular update to the Editor. It will send the update itself. If it does not have the latest update, the reporter which has the latest update will send it to the Editor.

Calculation showing number of MPI Calls:

Assumptions:

1.numNews = 10 =====> Original Content is only 10 news. Rest all are updates to one of these 10 news.(This value doesn't matter much for the sake of calculation).
2.TOTAL_NEWS - need not be fixed, the final calculation will involve TOTAL_NEWS.Used as X in calculation
3.There is only one Editor.
4.numReporters - need not be constant. The final calculation will involve numReporters. Used as R in calculation

5. Mapping of reporters to newsId is random. But if there are X original news. X/2 of them will be shared between reporters(how many editors will be told later) and X/2 will be exclusive news. Also, in the X/2 news that is shared. Each of the shared news will be shared roughly among 3 Reporters. (For the calculation of the expression, these assumptions hold, in reality the allocation is random but more or less matches the above said assumptions).

BROADCAST COST: Initially the source sends a broadcast to all the other processes involved to share the mapping related data. If there are a total of R Reporters. Then the number of MPI_Send calls used will be R+1. Same holds true for the MPI_Recv Calls. So, 2*(R+1) blocking send and receive calls. BROADCAST COST = 2*(R+1).

EXCLUSIVE NEWS COST: Assuming out of total of X news generated, X/2 are exclusive. The source will have X/2 MPI_Isend calls, the reporters will have X/2 MPI_Recv calls. Once received the reporters will forward this news to the Editor directly, because it is exclusive. This has X/2 MPI_Isend calls and X/2 MPI_Recv calls. A total of X MPI_Isend calls and X MPI_Recv calls will be used.EXCLUSIVE NEWS COST = 2*X.

SHARED NEWS COST: There are X/2 shared news. Each shared news is shared between 3 reporters. The Source will send each of the shared news to all the 3 reporters involved. So the source will have a total of X/2 * 3 MPI_Isend calls. Each of the sent message is received by the reporters. So, a total of X/2 * 3 MPI_Recv calls. Now, Reporter Reporter Communication related calls. In the Worst case, Each of the involved 3 reporters will send a message to the other two to check latest or not. So a total of 6 messages

will get exchanged for one shared news. For X/2 shared news, there will be (X/2)*6 MPI_Isend and X/2*6 MPI_Recv calls. Then one of the reporter will send the latest one to the editor. So, a total of X/2 MPI_Isend calls and X/2 MPI_Recv calls are used. SHARED NEWS COST: (X/2)*6 + (X/2)*3.
So, the final number is : 2*((R+1) + X + ((X/2)*3) + ((X/2)*6) + (X/2)) MPI calls are being made. Multiplication by two is because every Send has a corresponding Recv.

BNPM:

Single Source used. Number of Editors to be set using the variable numEditors, and number of Reporters using the numReporters variable. The source is process 0, processes 1 to numEditors are Editors and processes numEditors+1 to end are reporter processes.

Each editor has specific set of reporters assigned to him. This mapping is done by the source and is communicated to all the processes in the beginning of the program. This mapping will be used in the future.

Each editor receives a news update from the set of its reporters. The Editor has an EditorNewsMapping which the source sends to it. Using this information the Editor will know the news received is a shared news or exclusive. If exclusive, it will straightaway publish it. Else, it will communicate with the other Editors and come to a consensus as to who should publish.

Source: Same as in the case of BNPS, except it does some extra mapping calculations and send the mapping to all processes.
Reporters: A Reporter has a specific editor to which it maps to.
Editor: Almost same as in BNPS, except When there is shared news, the Editor communicates with concerned other reporters and comes to a consensus.

Calculation showing number of MPI Calls:

Assumptions:
1.numNews = 10 =====> Original Content is only 10 news. Rest all are updates to one of these 10 news.(This value doesn't matter much for the sake of calculation).
2.TOTAL_NEWS - need not be fixed, the final calculation will involve TOTAL_NEWS.Used as X in calculation
3.numReporters - need not be constant. The final calculation will involve numReporters. Used as R in calculation

4.numEditors - need not be constant. The final calculation will involve numEditors. Used as E in the calculation.
5. Mapping of reporters to newsId is random. But if there are X original news. X/2 of them will be shared between reporters(how many editors will be told later) and X/2 will be exclusive news. Also, in the X/2 news that is shared. Each of the shared news will be shared roughly among 3 Reporters. (For the calculation of the expression, these assumptions hold, in reality the allocation is random but more or less matches the above said assumptions). Editor to Reporter mapping is deterministic. Reporter R will be mapped to Editor E if(R%numEditors == E). That is Reporters are almost equally allocated among editors.
6.Given Editor-Reporter and Reporter-News Mapping, a Editor-News Mapping can be calculated. Assuming that on average Editors have 3 common news, calculation has been made.(news here refers to the original news, if original news is 10 and 3 are shared, all the updates of this 3 shared news are also shared).

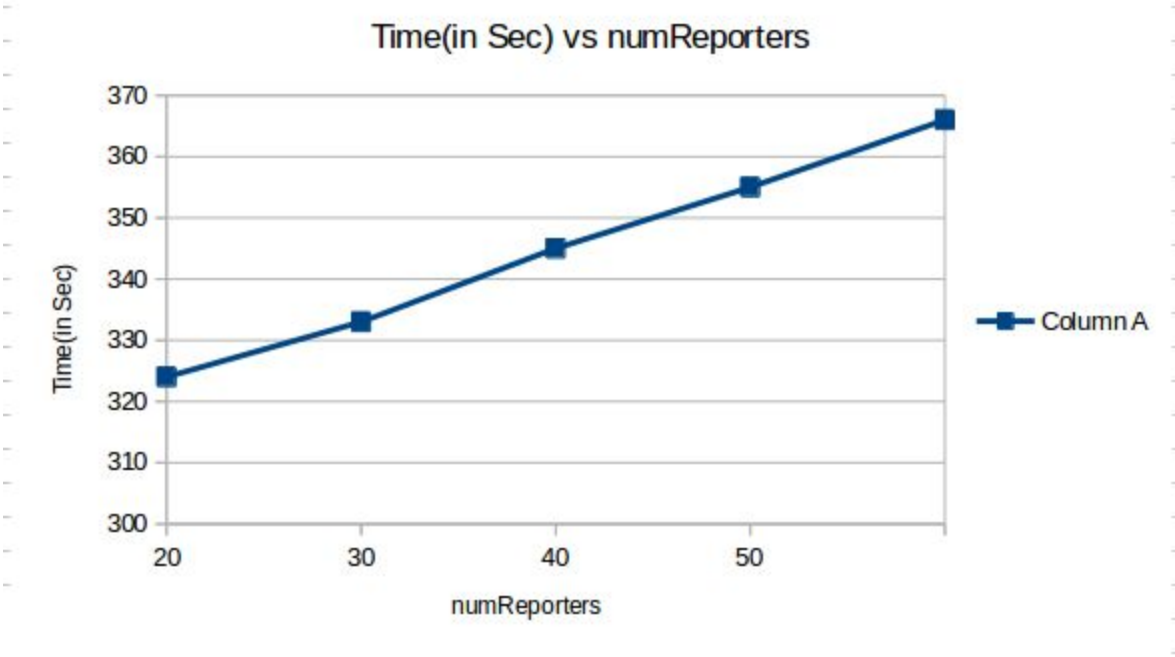
BROADCAST COST: Initially the source sends three broadcast to all the other processes involved to share the mapping related data. If there are a total of R Reporters and E Reporters. Then the number of MPI_Send calls used will be R+E. Same holds true for the MPI_Recv Calls. So, 2*(R+E) blocking send and receive calls. BROADCAST COST = 2*(R+E).

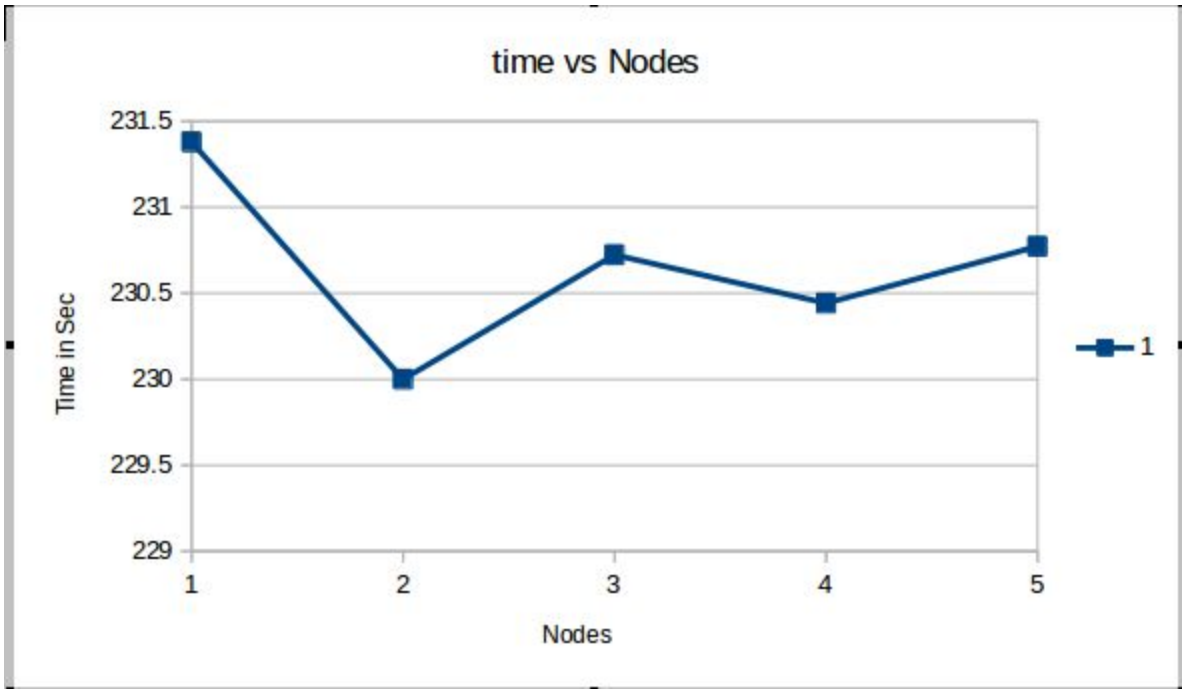
EXCLUSIVE NEWS COST: Assuming out of total of X news generated, X/2 are exclusive. The source will have X/2 MPI_Isend calls, the reporters will have X/2 MPI_Recv calls. Once received the reporters will forward this news to the Editor directly, because it is exclusive. This has X/2 MPI_Isend calls and X/2 MPI_Recv calls. A total of X MPI_Isend calls and X MPI_Recv calls will be used.EXCLUSIVE NEWS COST = 2*X. This is same as the BNPS case.

SHARED NEWS COST: There are X/2 shared news. Each shared news is shared between 3 Editors. Assuming within the reporters of an editor there is no sharing, no of calls made will be 2*(X/2). Now comes in the Editor Calculation. 6*E messages will be transmitted between editors.

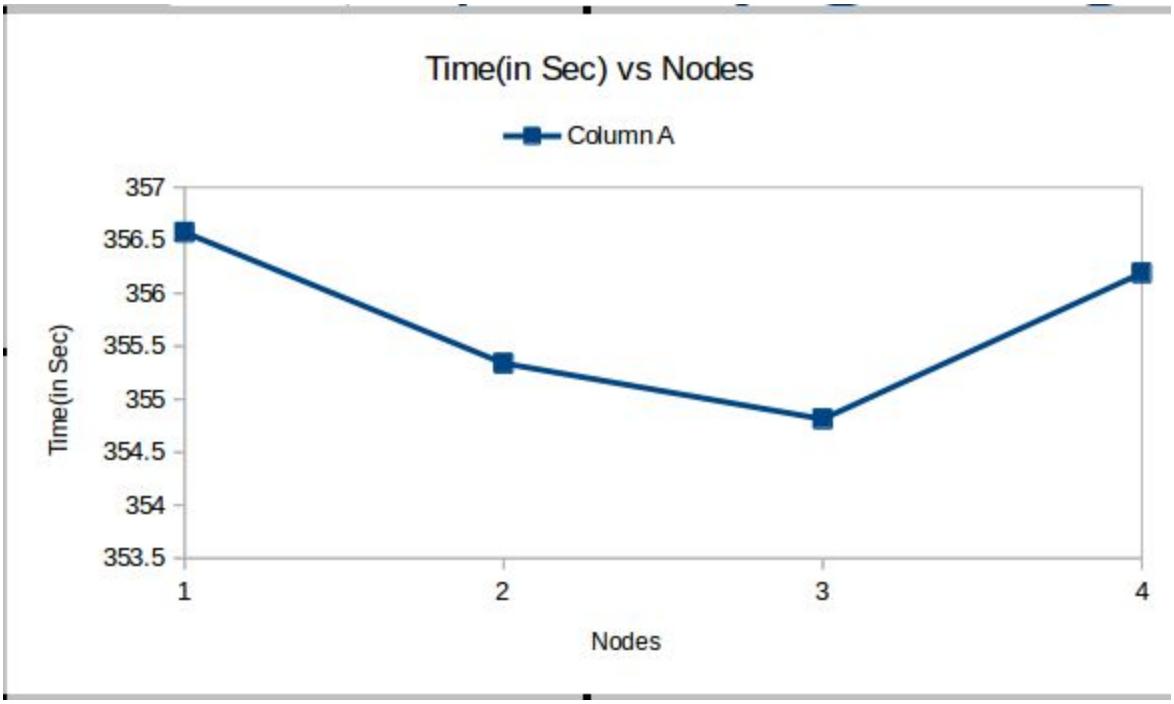
Final Calculaion: 2*(R+E) + 2*X + X + 6*E.

Profiling: BNPS





BNPM:


Time(in Sec) vs Nodes
Column A
357
356.5
356
355.5
355
354.5
354
353.5
Time(in Sec)
1
2
3
4
Nodes