MANDELBROT REAL TIME ZOOMING IN DSM ARCHITECTURE

AUTHOR: PRAKASH NAIR
PROFESSOR: DR. RUSS MILLER
DATE: NOV 30, 2010
Scope of the Presentation

- Define Mandelbrot set and the zooming feature
- Algorithm for sequential and parallel approach
- Implementation
- Results
- Screenshots
  - Environment: 2 Nodes x 2 Physical processor
  - 1 Physical processor → 2 logical cores (intel hyper threading)
- Limitations and Future Scope
- References
The Mandelbrot set is a mathematical set of points in the complex plane, the boundary of which forms a fractal.

\[ z_{n+1} = z_n^2 + c \]

- Iterations: Multiple of 400
- Bail out: 2
- For Julia Set, keep ‘c’ constant.
APPROACHES FOR DIFFERENT ARCHITECTURE

- SISD ARCHITECTURE
- SMP ARCHITECTURE
- MPP ARCHITECTURE
- DSM ARCHITECTURE
Sequential Algorithm

START

Initialize

Call Mandelbrot

Call Display

Quit

STOP

Event: MousePress

Co-ordinate Points Zoom Value
DSM Algorithm

START

Initialize

Call Master Node

Call Data Node

Call Display

Call Mandelbrot

Event: MousePress

Quit

STOP

Co-ordinate Points

Zoom Value
Mandelbrot Pseudo Code

- Plot is of 400x400
- Range_of_points are the selected points, which needs to be extrapolated onto the 400x400 plot
- History Data Type and Zoom variable
- Converting \((x,y)\) co-ordinate points to cartesian co-ordinates on complex plane.

\[
\begin{align*}
c_{\text{re}} &= \text{MinRe} + x \ast (\text{MaxRe} - \text{MinRe}) / (\text{ImageWidth} - 1); \\
c_{\text{im}} &= \text{MaxIm} - y \ast (\text{MaxIm} - \text{MinIm}) / (\text{ImageHeight} - 1);
\end{align*}
\]
Mandelbrot Pseudo Code (cntd)

- Master Node triggers the Data Nodes (Broadcast)
- Split the plot amongst number of processor
- Split the range_of_points amongst number of processor
- For each Processor find,
  - Points with minimum and maximum number of iterations
- Store the iteration count for each points in the plane
Mandelbrot Pseudo Code (cntd)

- Call the Reduction mechanism for collecting the points with minimum and maximum iteration
- Call the Gather mechanism, to collect the iteration count for all the points
- Master Node calls the Display method
- In Display method, divide the plot amongst the threads
- In U2, intel Xeon, we have 2 physical cores and 4 logical cores (HT). Thus, Number of threads initialized is 2 - 4
Mandelbrot Pseudo Code (cntd)

- MPI_Reduce()
- MPI_Gather()
- MPI_Bcast() {As a triggering event}
- Grep –c “processor” /proc/cpuinfo
- OMP for,critical,section
- GMP
Implementation

Mandelbrot
Calculate_set()

CallByRoot
Mandelbrot()

Qapplication
Mandelbrot()
Display()
Trigger_CallByRoot()

Widget
Object GridLayout
Object PushButton
Object CannonField
Object RadioButton
Object LCDScroll

Qapplication()

Main
If(Master) {
Call Widget()
}
Else{
Call CallByRoot()
}
Results

U2

Zoom

seconds

Initial zoom1 zoom2 zoom3

DSM MPP
Screenshots (Initiate)
Screenshots (Zoom 1)
Screenshots (Zoom2)
Screenshot (Julia set Initiate)
The zooming beyond 7x does not show significant insight. More analysis required on this.

Though the sequential algorithm is optimal, the re-engineering for the MPI version could have better

More arenas for using OpenMP
References

- http://doc.qt.nokia.com/3.3
- Wikipedia.org
- Notes by Prof M. Jones
- Multi-core Faculty Training_v21.01_Student Workbook (Intel OpenMP 3.0)
- MPI:The Complete Reference –Vol 1.0
- GMP:GNU multiple precision library