

Grid-enabled Sawing Optimization: from scanning images to cutting solution

By Zhifeng Yun





Outline

- Introduction
- TOPSAW procedure
- Limitation
- DA-TC
- Results
- Conclusion



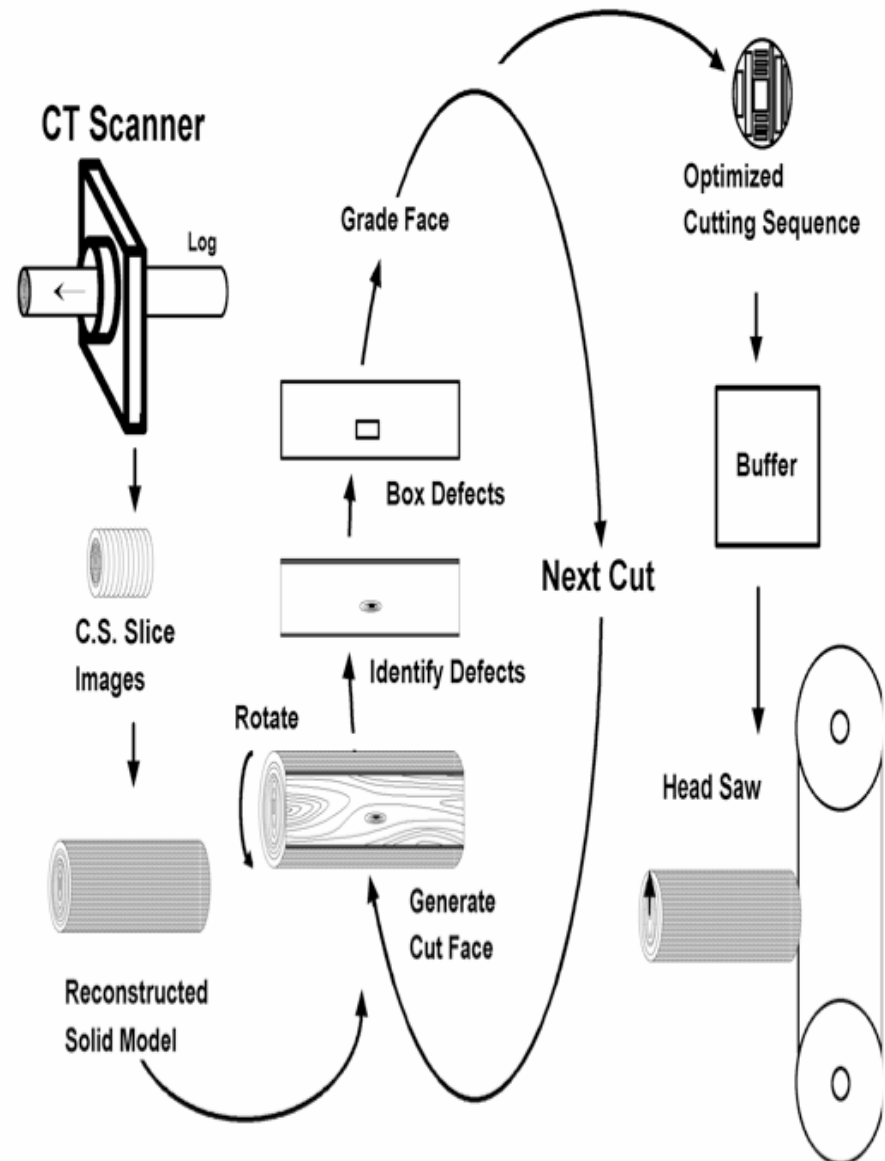
Introduction

- Hard to detect defects externally
- Unrepeatable sawing
- **TOPSAW**: Trainning and Optimization system for SAWing Logs
 - replicate the sawmill cutting
 - determine the optimal cutting pattern to maximize the value of hardwood lumber produced



TOPSAW Procedure

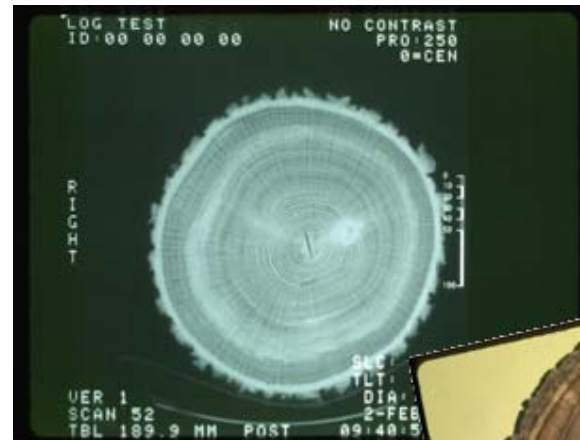
- Scan a log
- Reconstruct 3-D virtual log
- Identify the internal defects
- Box the defects
- Get optimal sawing sequence





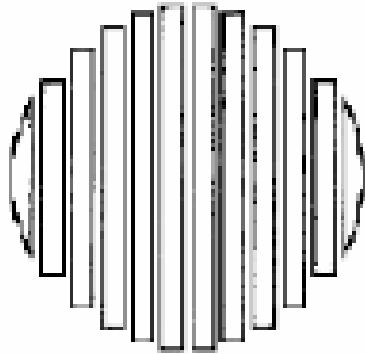
CCT Log Scanning & Defect Detection

- 5 o'clock:
a scar tissue
- 2 o'clock:
a big knot
- 10 o'clock:
a trace of a knot
- Center:
a heart check

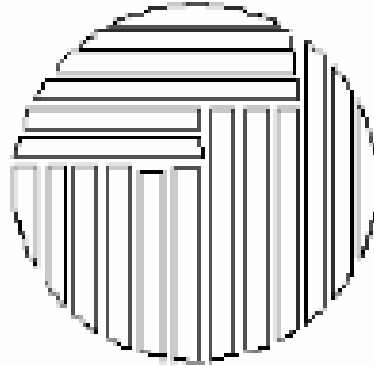




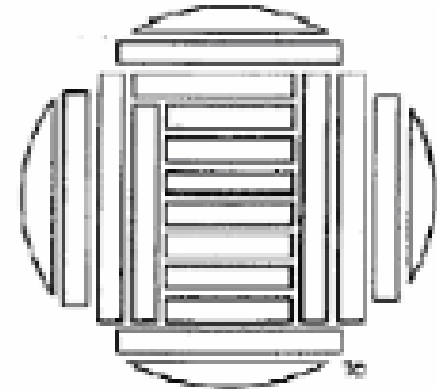
CCT Log Sawing



Live Sawing



BI-Directional Sawing



Grade Sawing

Making optimal cut involves:

- The optimal rotational orientation
- The optimal depth of opening cut

 CCT Limitation

- Grade sawing can increase the value of the lumber produced by 18% over actual sawmill production
- However, with 6 different log orientations and 17 different depths of opening cut in each one of the 4 sides, there are 501,126 ($6 \cdot 17^4$) combinations
- It will take several days to optimize the combinations
- Solution: Grid enabled calculation
- DA-TC model further reduced the execution time



CCT Current Issues

- Wait for execution completed after submitting tasks to the end of queues of remote clusters.
- Problems arise:
 - Bottleneck: slow cluster
 - Poor load balancing
 - Long turnaround time
 - Lack of effective user-execution interaction
 - Difficulties for status monitoring



CCT DA-TC: New execution model

- Dynamic Assignment with Task Containers
 - Dynamic load balancing: assign tasks to remote site based on runtime status.
 - Task container technology:
 - Waiting for resource allocation
 - Holding resources
 - Managing the lifetime of task execution
 - Responding user's requests
 - Collaborating inter-task communications
- Two major components
 - Application execution agent (AEA)
 - Task containers (TC)



- Execution preparation
- Dynamic task assignment
- Execution termination
- Monitoring application progress, task container status, and task status

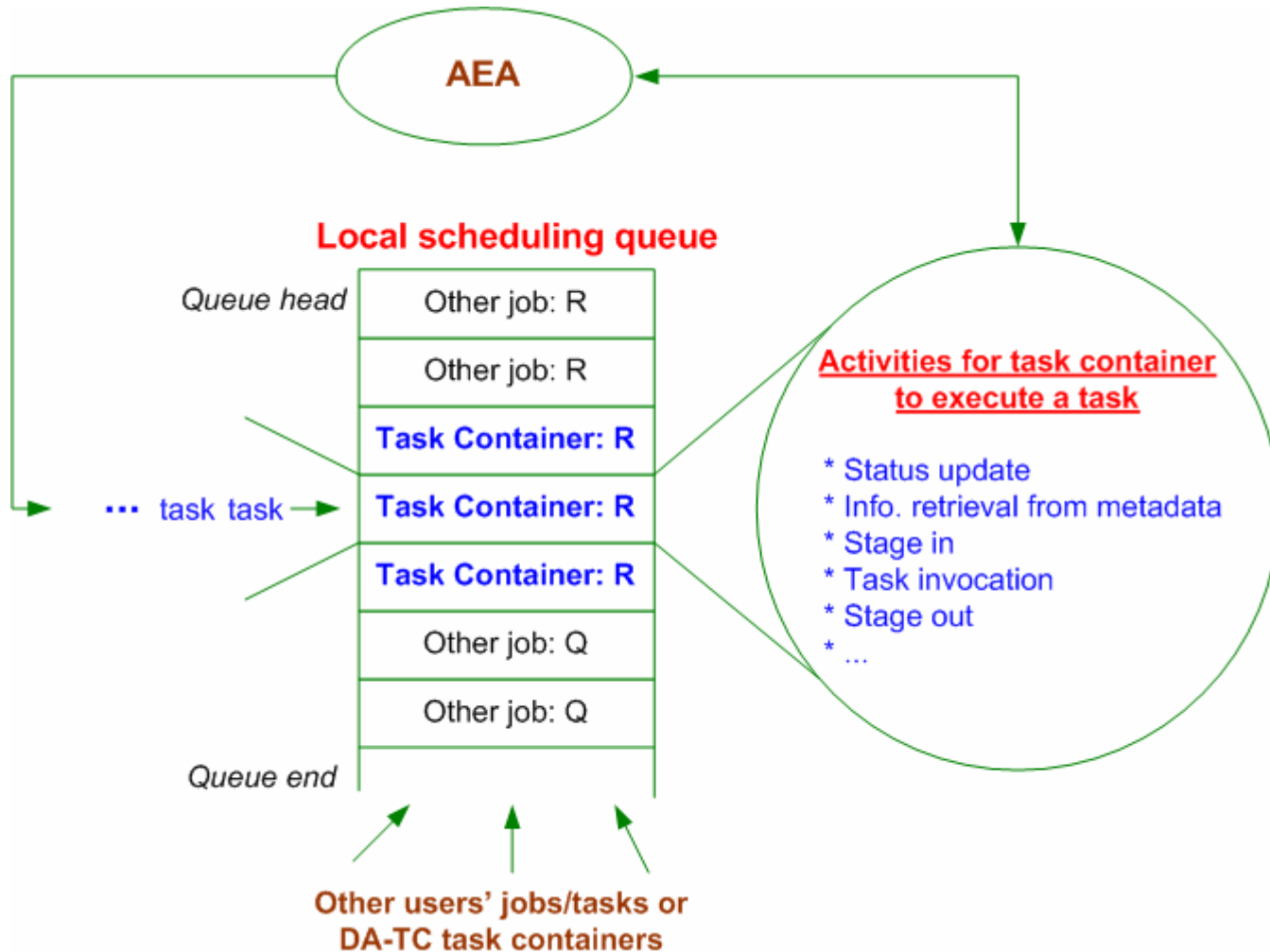


- A TC is viewed as a normal job
- A TC retrieves task execution requirements from metadata:
 - task dependences
 - executable location
 - data location, etc
- AEA monitor and steer the assigned task(s) in a TC

```
while (TRUE)
{
    if (application termination signal)
    {
        exit 0;
    }
    else
    {
        update the status;
        get next task metadata from AEA;
        retrieve information from metadata;
        task execution (stagein, invocation, stageout, etc);
        update the status;
    }
}
```

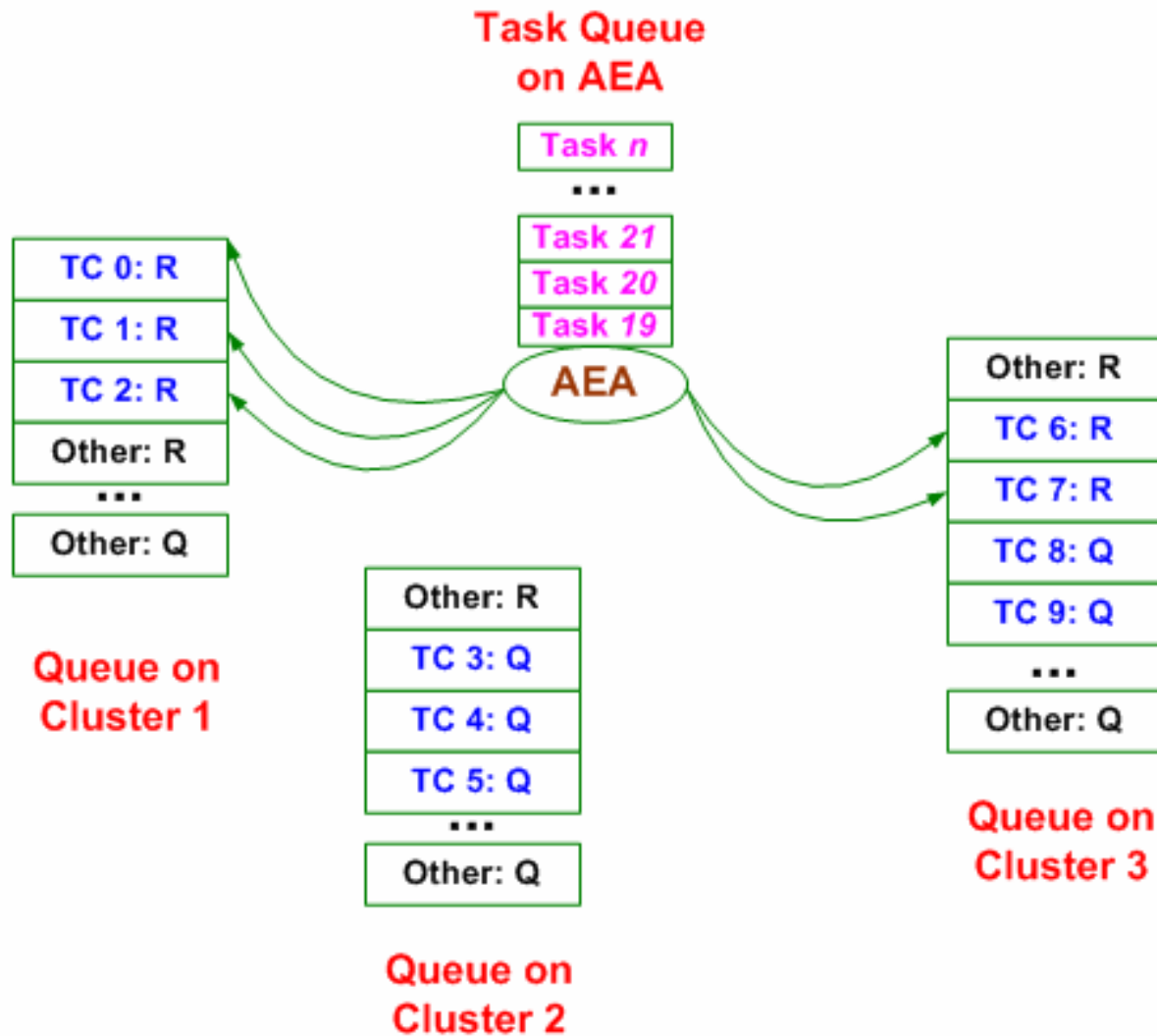


CCT AEA and TC





Runtime scenario





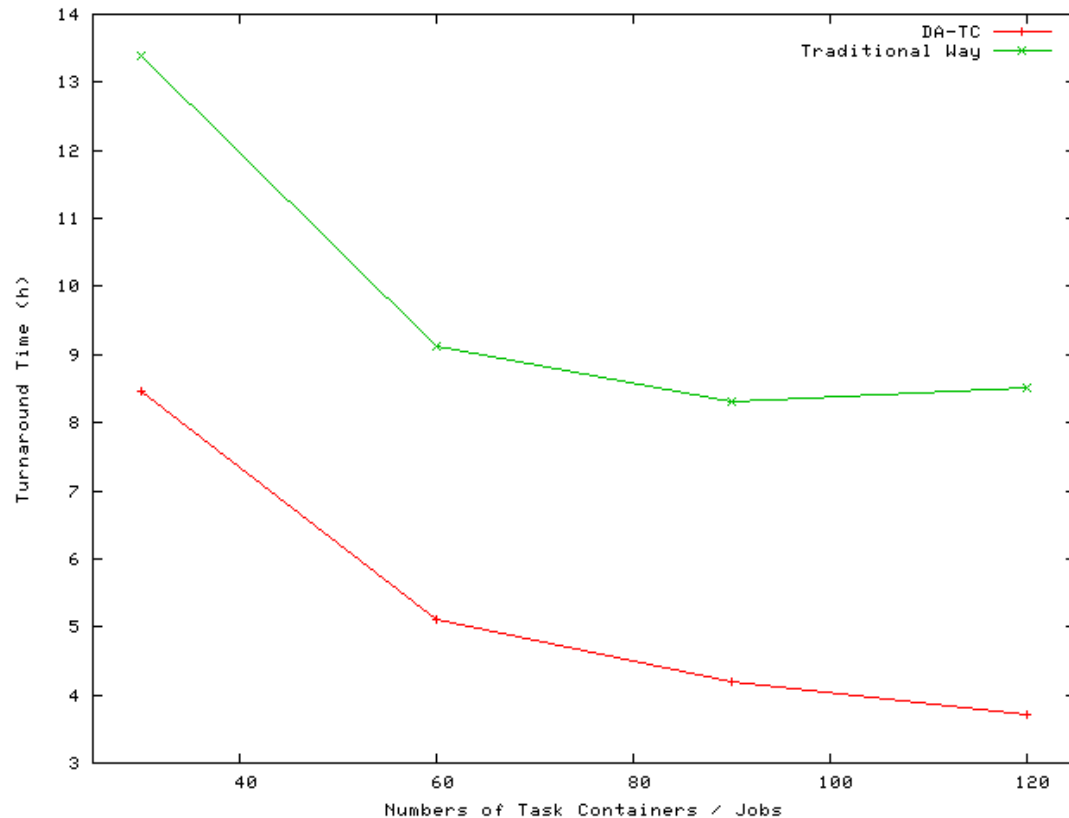
CCT DA-TC Advantage

- Dynamic load balancing – resource utilization
- Significant reduction of turnaround time
- Slow resources: beneficial factors, not bottleneck any more
- Easy for user-execution interaction
- Easy to monitor / steer execution progress
- Extensive applicability

Results(1)

An identical application submitted with different number of task containers.

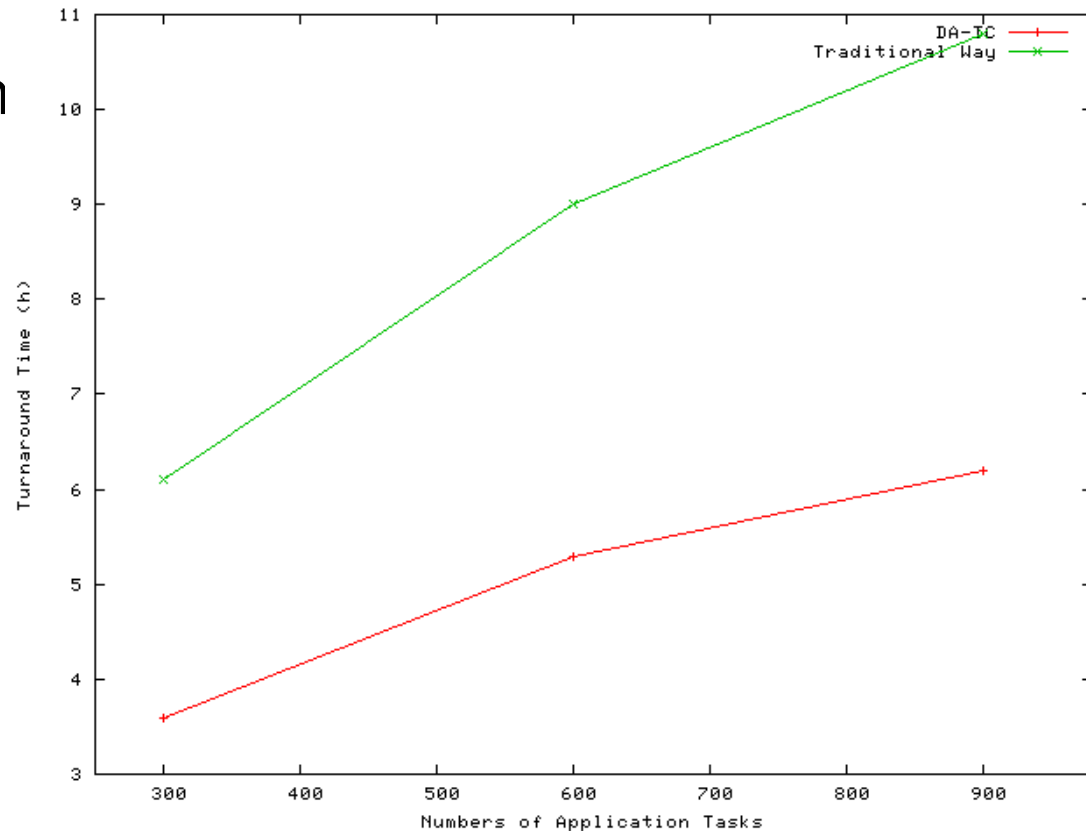
- Application consists of 500 tasks
- Submit with 30, 60, 90, 120 TCs.
- Turnaround time can be decreased by increasing the number of task containers.
- Slow participating cluster can be the bottleneck



Results(2)

The number of task containers / jobs fixed with different size of applications.

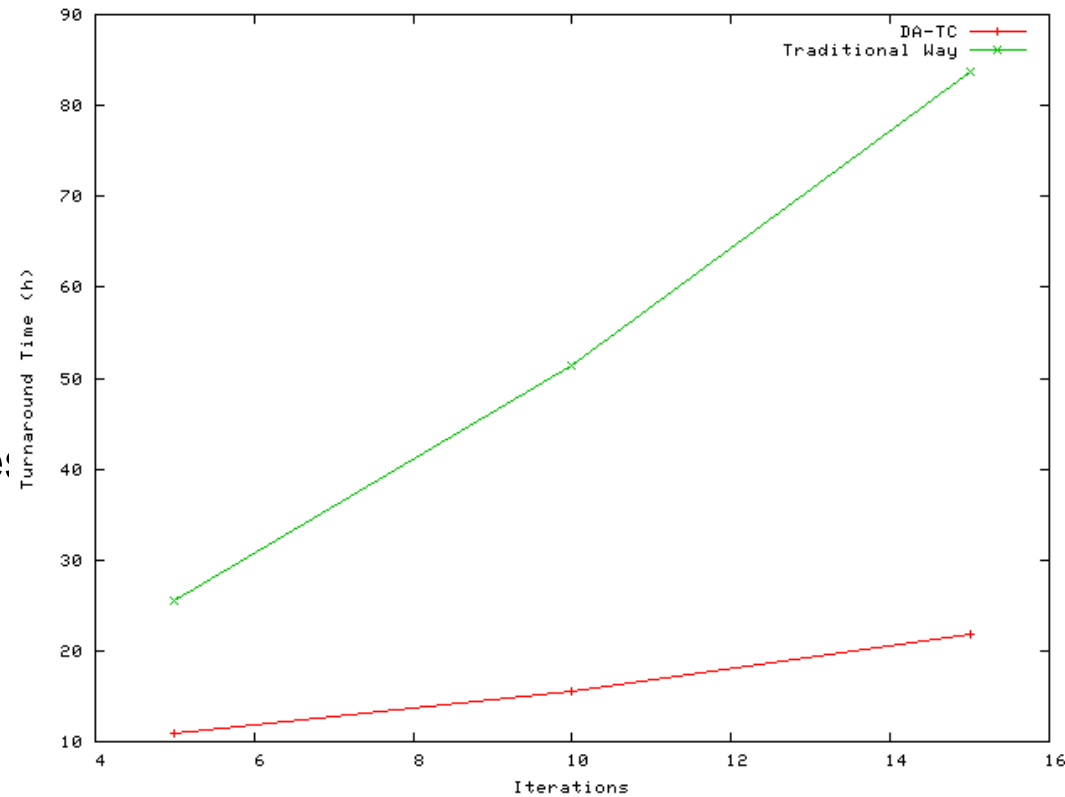
- Number of TC is 60
- Three applications with 300, 600, 900 task numbers
- The turnaround time is greatly reduced if applications with different size are submitted via DA-TC



Results(3)

The number of task containers / jobs fixed with different number of iterations

- Number of TC is 20
- Three inverse modeling processes with 5, 10, and 15 iterations, each iteration has 100 tasks
- Turnaround time increase, much faster under traditional way
- TCs never release the resources until the end VS tasks submitted to the end of the local scheduling queues in each iteration



 CCT Conclusion

- By using the CT scan technique, a virtual log can be reconstructed and the best cutting solution can be calculated before actual cutting
- A average of 15% gain over sawmill production can be obtained by using TOPSAW
- Grid enabled calculation makes it possible to fit the industry requirement
- DA-TC execution model further reduced the execution time
- The project is still in progress, more features are expected to add into the system

