Towards a Framework for Building Maneuverable Applications

William Clay Moody
wcm@clemson.edu
Amy Apon
aapon@clemson.edu

Clemson University

Abstract

Maneuverable applications are distributed and parallel applications taking advantage of the addition and removal of resources within the computing system, giving the perception of movement. These resources can be computation, network, storage, or the application itself. The perceived movement of the maneuverable application is deliberate and skillful with the desire to gain system advantages. Some of those advantages are provisioning resources, optimizing applications, and improving cyber security This poster describes our work to date on building, designing, and modeling maneuverable applications within shared computing resources. Furthermore, we discuss future applications on maneuverability and its potential enhancement.

Background

Distributed and parallel systems allow multiple interconnected computers to perform synchronized computations. These systems allow applications to scale to upwards of thousands of processors. Much interest exists in continual improvement of the performance of these systems to provide increased return on investment.

The Department of Defense defines maneuver as "A movement to place ships, aircraft, or land forces in a position of advantage over the enemy."[1] Besides a malicious attacker, distributed systems have adversarial conditions such as mechanical failure, network imbalance, disastrous weather, or scheduled downtown.

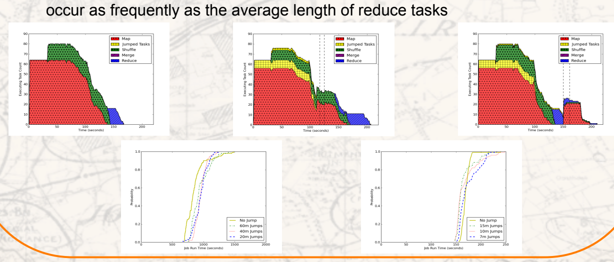
References

1. Joint Chiefs of Staff, Joint Pub. 1-02, Department of Defense Dictionary of Military Terms and Associated Terms (8 Nov. 2010)
2. Applegate, S.D., "The principle of maneuver in cyber operations," Cyber Conflict (CYCON), 2012 4th International Conference on , vol., no., pp.1,13, 5-8 June 2012
3. Moody, W.C. et al; "JUMMP: Job Uninterrupted Maneuverable MapReduce Platform", Cluster Computing (CLUSTER), 2013 IEEE International Conference on , 23-27 Sept. 2013
4. Moody W.C., et al; "Reconfigurable Network Testbed for Evaluation of Data Center Topologies", to appear: The 6th International Workshop on Data-intensive Distributed Computing (DIDC'14) , 23-27 Jun. 2014
5. Moody, W.C., Apon, A., "Modeling Defensive Maneuverable Cyber Platforms with Petri Nets", Working Paper, 2014

Maneuver for Resource Provision

JUMMP: Job Uninterrupted Maneuverable MapReduce Platform [3]

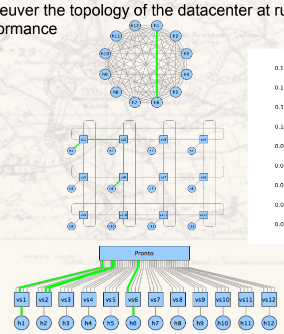
- Enables the integration of Hadoop into the existing large scale computational infrastructure
- Supports high availability and continuous computing for research and education
- Incurs no additional financial or administrative overhead
- Interactive pseudo-persistent MapReduce platform within the existing administrative structure of an academic high performance computing center
- As efficient as a persistent Hadoop cluster on dedicated computing resources, depending on the jump time
- Cluster remains stable, with good performance, in the presence of jumps that occur as frequently as the average length of reduce tasks

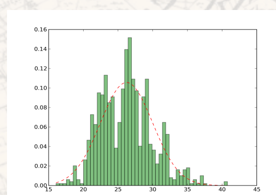


Maneuver for Application Optimization

FORCE: Flow Optimized Route Configuration Engine [4]

- Reconfigurable testbed with SDN hardware, a limited number of workstations, and custom software to emulating the behavior of datacenter
- Use the testbed to study the effects of network topologies on distributed applications
- Use the testbed to get an early indication if SDN-enhancement hypotheses are worthy of further investigation
- Maneuver the topology of the datacenter at run-time to optimize application performance

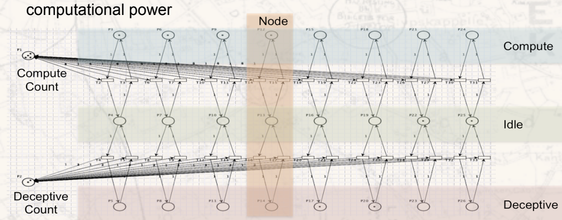




Maneuver for Security Improvement

Defensive Cyber Petri Net Model [5]

- Four Basic Forms of Defensive Cyber Maneuver [2] include Perimeter Defense, Moving Target Defense, Deceptive Defense, Counter Attack
- Integrating Moving Target and Deceptive Defense into a distributed application framework can provide improved security
- Petri Net model of a maneuverable system where nodes either can be in computation, idle, or deceptive states
- Nodes constantly moving between states, maintain minimum levels of computation and deceptive states
- Distribution of nodes in each state depending on the current threat level
- Evaluate trade-off between improved security and survivability versus loss in computational power



Future Work

- We are currently further investigating the preliminary results observed with Hadoop shuffle traffic with random topology placement. We are designing an intermediate service to proactively reprogram the testbed network after map and reduce task assignments, but before execution of the tasks begin. We hope to achieve speedups in Hadoop job performance even when considering the additional overhead of predictive reconfiguration.

- We are using JUMMP in semester projects for undergraduate distributed computing courses. We are extending the deployment of JUMMP with an internal scheduler, removing the need for PBS or other shared resource schedulers. Additionally, we are adding a visualization layer to the software to allow users and administrators to see current and past node utilization by JUMMP.

- We plan to expand the Petri Net model to include models of the actions of attackers. Our research goals include studying the impact on time to failure and time to compromise when a system is maneuvering at various speeds and ratios of computation vs. deception.

Timeline

August 2014:	Dissertation Proposal
March 2015:	Dissertation Defense
May 2015:	Graduation

Partially Funded via United States Army and United States Military Academy

Partially Funded via National Science Foundation grant #1251544