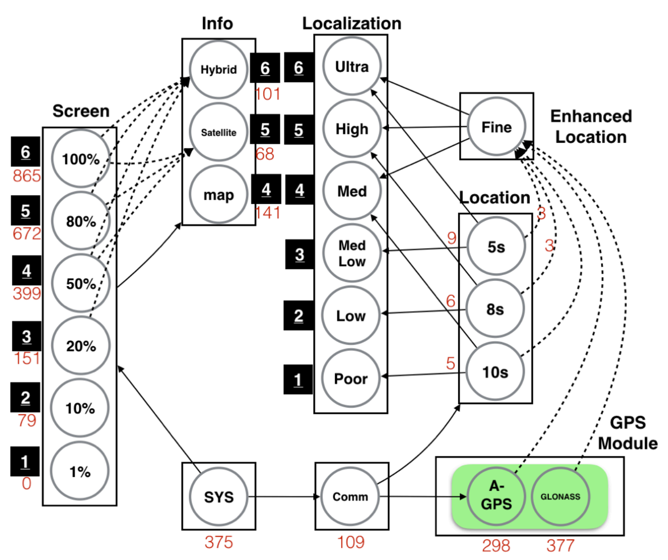
# RUTGERS CSR:Small: Energy-Aware Redundancy Management

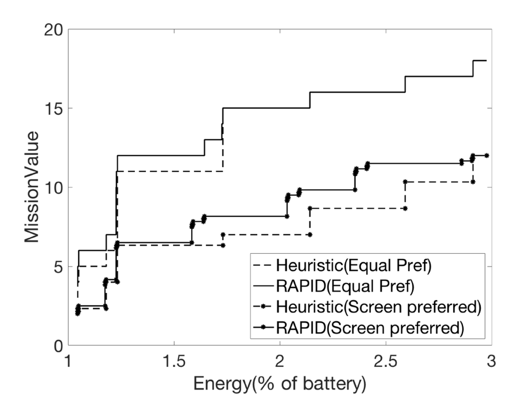
PI: Ulrich Kremer NSF-CSR #1617551

EEL Lab
Computer Science

### RSDG NavApp on Smartphone

Navigation application on Android that balances multiple services to meet the energy requirement.

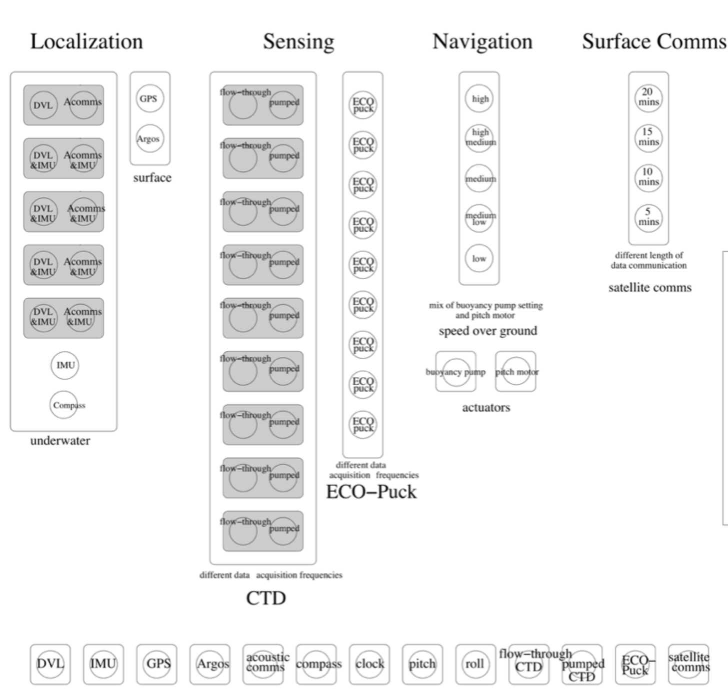


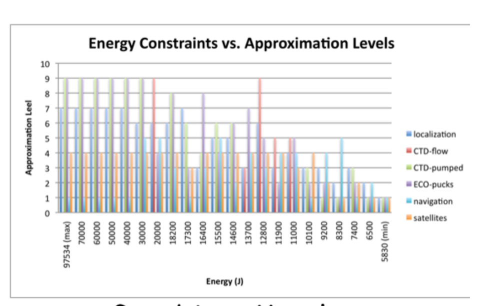


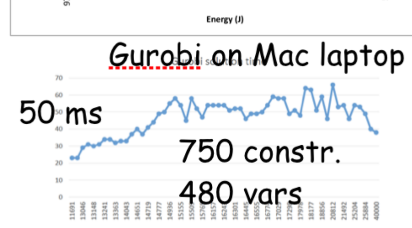

Productivity profiles of RAPID vs. heuristic with different user preferences:
top: 6:6:6 (1:1:1), bottom 10:1:1

### RSDG Underwater Glider (Slocum)

Multiple sensors and dependencies between services (dep. edges not shown)

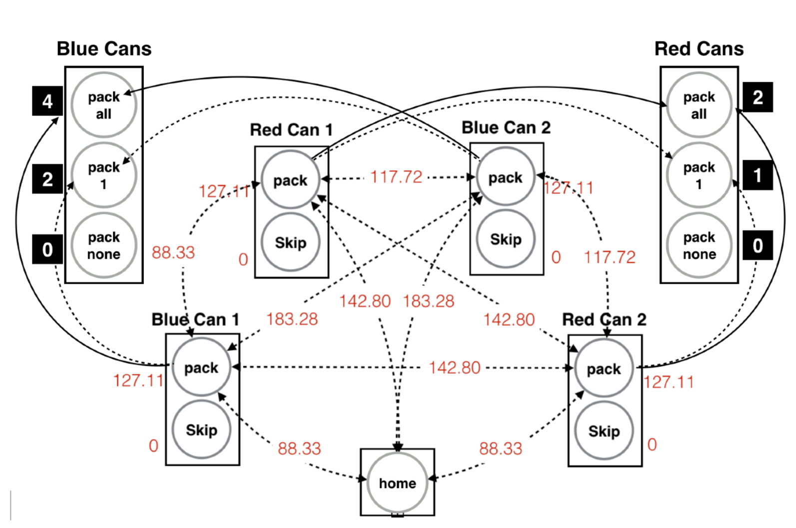


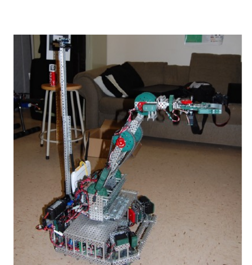



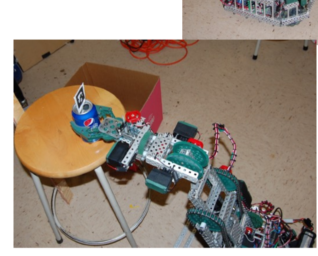



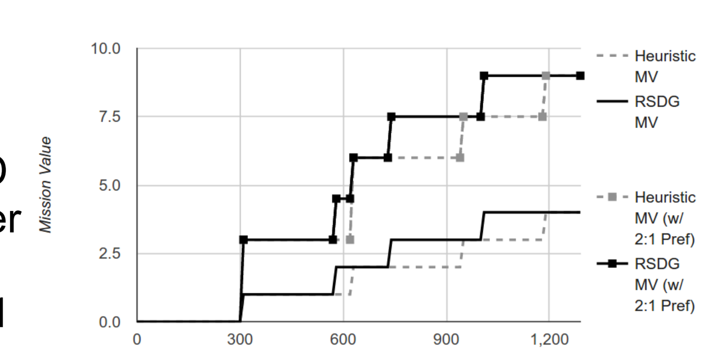

### RSDG Service Robot

Expressiveness of RSDG to encode route planning and service selection problem.








Productivity profiles of RAPID vs. heuristic with different user preferences:
top: 3:1.5 (2:1), bottom 1:1

## Target Applications

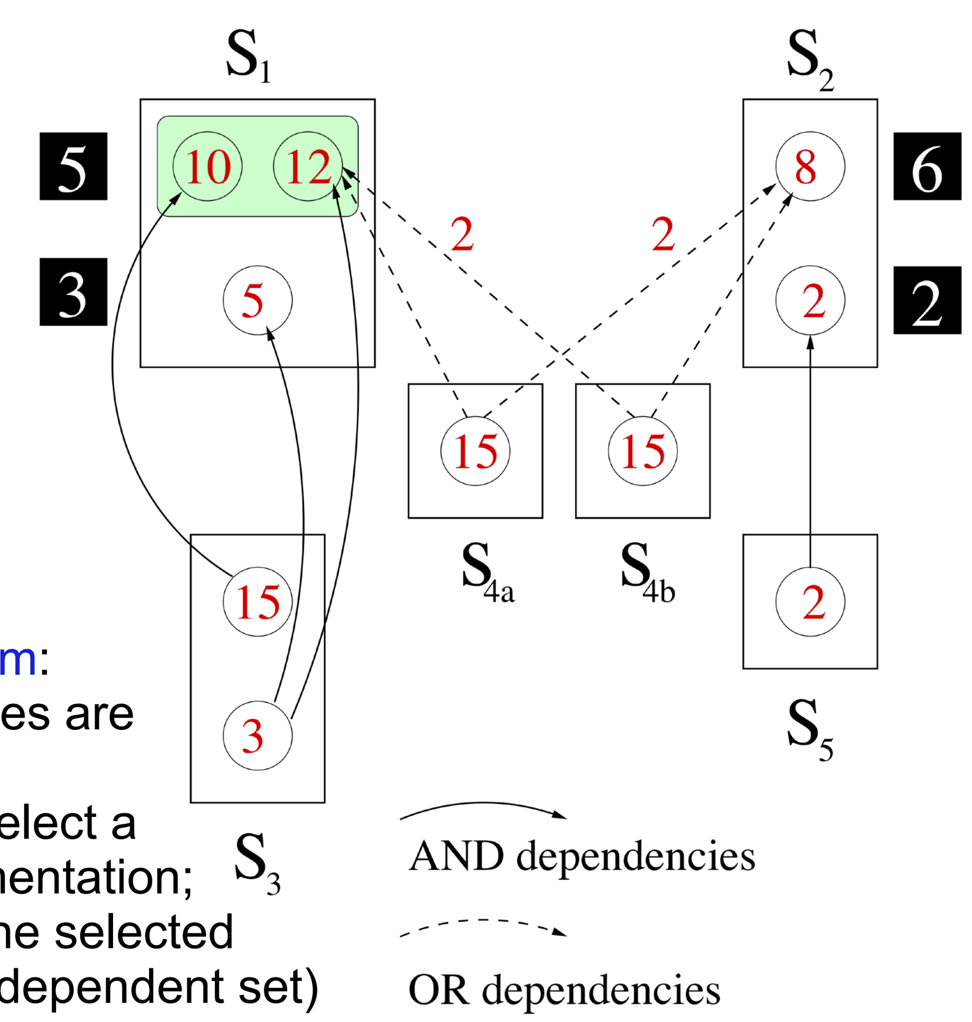
1. Applications are composed of sets of collaborating services (producer- consumer, resource sharing)
2. Each individual service may provide discrete levels of quality vs. energy (resource) tradeoffs.
3. Nothing is free! User wants best overall application outcome within a given energy budget.
4. Need to deal with failure and uncertainty.
5. Need to support user customizable quality metrics.

## RSDG (Redundant Services Dependence Graph)

3 types of redundancy:
- approximation
- implementation
- replication

mission value

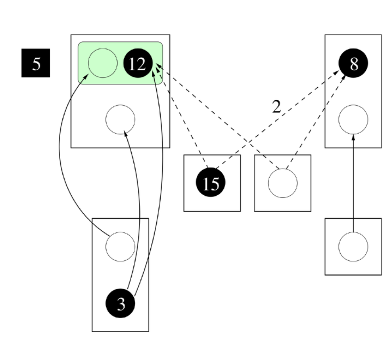
energy consumption



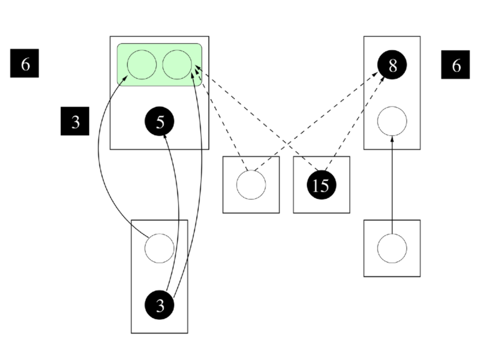

RSDG service selection problem:

- Services with mission values are considered (user) critical
- For each critical service, select a single service level and implementation; do the same for each service the selected service depends on (transitive dependent set)
- Sum of mission values of critical services is maximal under given energy constrains; energy is minimized for this maximal mission value

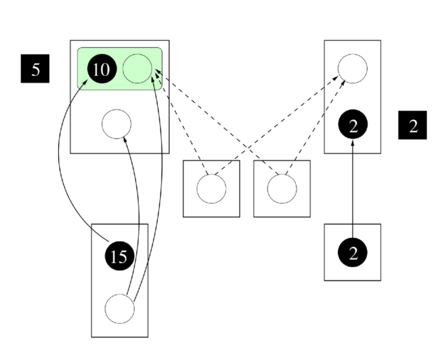
## Service Selection and Productivity Profile



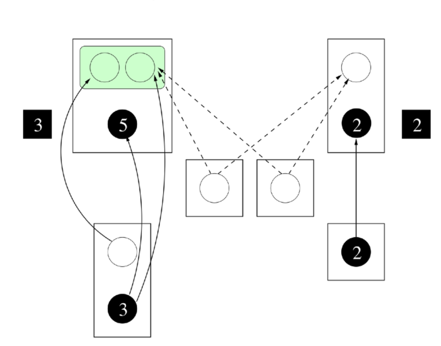
Solution for energy = 40 and example mission values: 11



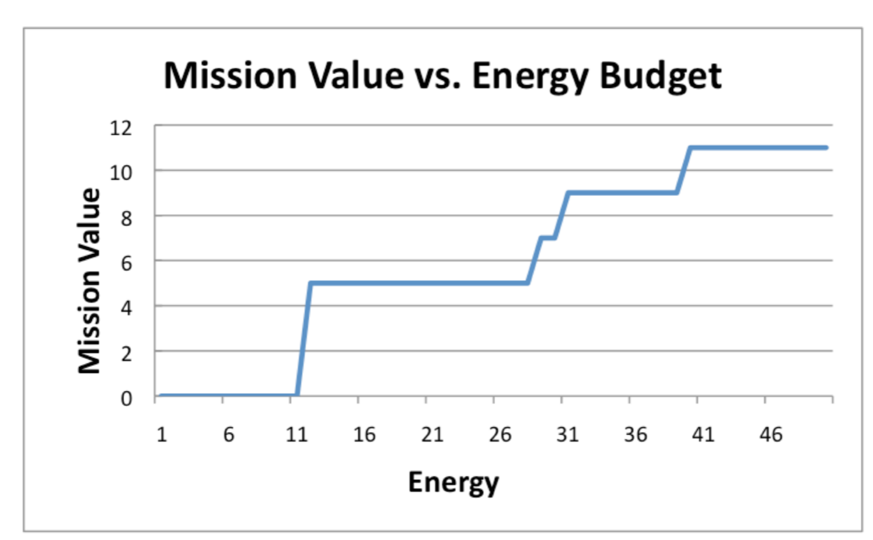
Solution for energy = 31 and example mission values: 9



Solution for energy = 29 and example mission values: 7



Solution for energy = 12 and example mission values: 5




Optimal solution balances service qualities across the entire application; Solution seems sometimes non-intuitive. Solution is NP-complete;
Proof: Reduction from 3SAT.
Use of 0-1 integer programming formulation and Gurobi to solve the problem optimally.
Simulations can provide feedback during tradeoff space exploration via the Productivity Profile.

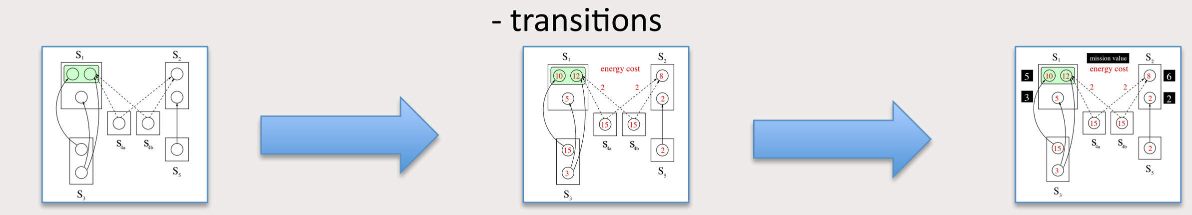
## RAPID System Overview

### Application Specification

**System Expert**
Services
Redundancy levels
Dependencies

**Automatic Tool**
Power dissipation of
- service levels
- communication
- transitions

**User**
Mission values (service priorities)
Energy budget & mission time
Environmental configuration

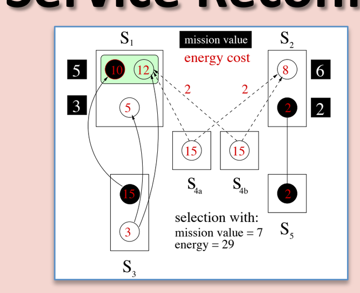


System RSDG → System RSDG with power annotations → Dual-weighted Full RSDG

### Dynamic Service Reconfiguration

Dynamic service selection problem
- maximize productivity
- minimize energy

Optimal and heuristic solutions

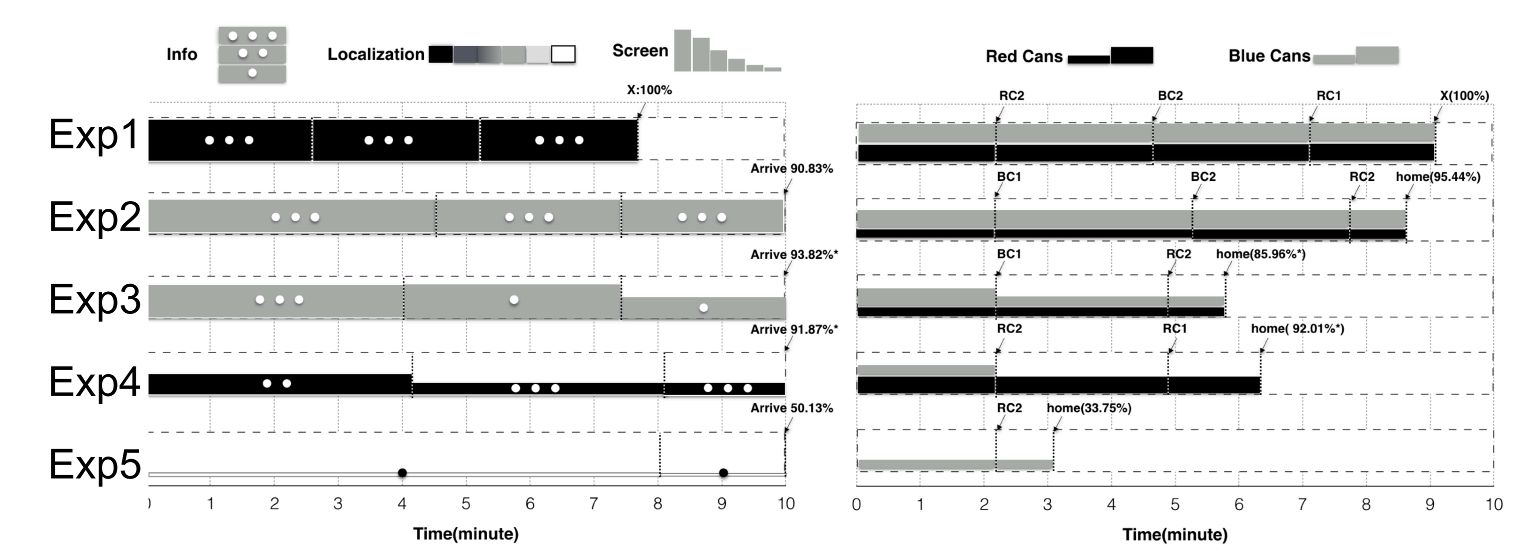


Dual-weighted full RSDG with service selection

Dynamic reconfiguration triggered by
- failure conditions
- changed energy budget
- environmental events
- fixed time interval

## Experiments

**Exp1**: Highest service levels w/o reconfiguration.
**Exp2**: RAPID with screen preference and stable energy consumption.
**Exp3**: RAPID with screen preference and unexpected energy drain.
**Exp4**: RAPID with localization preference and unexpected energy drain.
**Exp5**: Lowest service levels w/o reconfiguration but w/ unexpected energy drain.




**NavApp:** RAPID runs on LG Nexus 5 (Android), Quad-core, 2.3 GHz; Gurobi runs on remote server; receives 0-1 problem from phone, and returns result; power measured by Qualcomm's Trepn tool.
**Overheads**: 4 ms solution, 0.3 ms reconfiguration; 0.09% energy.

**Service Robot**: RAPID runs on ARM-based NVIDIA Jetson TK1 system; Gurobi runs remotely as in NavApp; power measured by on-board Pololu ACS711EX current sensor.
**Overheads**: negligible due to small problem size (>1 ms solve + reconfig); energy dominated by motors.

## Summary and Conclusion

A new RAPID framework explicitly models and represents redundancies through a graph representation (RSDG). This allows efficient search space explorations. The system expert provides the basic structure of the application using his/her domain knowledge. An automatic tool derives the hard cost metrics (energy) through online and offline training. The application user customizes the service preferences and specifies an overall energy budget. RAPID then adaptively balances the services configurations across the entire system to produce highest user productivity with lowest cost within the user-specific energy budget, even in the presence of failures and uncertainty. This is accomplished by reconfiguring the system in response to such unexpected events. **Future work** includes merging RSDGs and energy budgets of simultaneously executing applications.