Using Machine Learning to Control Coupled, Dynamical Life Support Systems

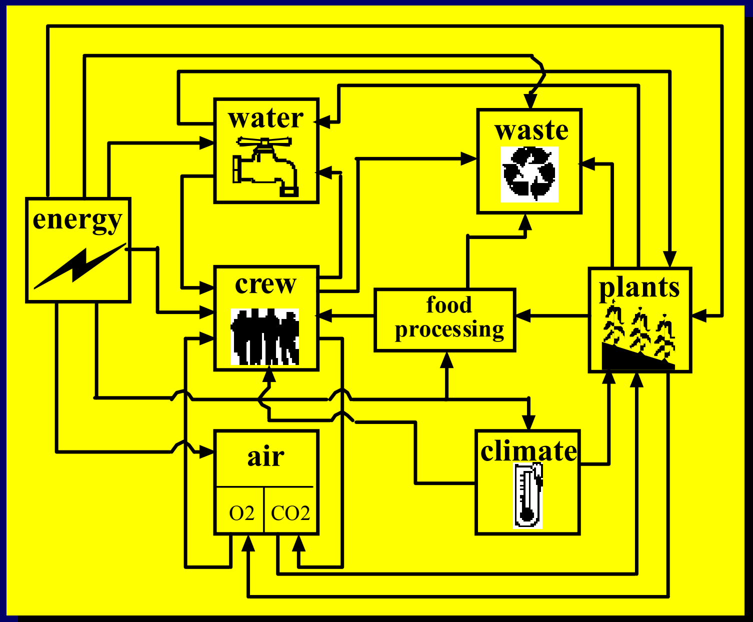
David Kortenkamp
NASA Johnson Space Center/Metrica Inc.
Houston TX 77058
kortenkamp@jsc.nasa.gov
http://www.traclabs.com/~korten

Team

- NASA JSC
  - David Kortenkamp, Pete Bonasso
- NASA Ames
  - Justin Boyan
- Rice University
  - Devika Subramanian
- Carnegie Mellon University
  - Jeff Schneider
- Naval Research Laboratory
  - Alan Schultz

Advanced Life Support Systems

- Regenerative
  - produce own food
  - recycle water and air
- Low margins, volume, mass, energy and labor
- Limited resupply
- Highly interconnected
- Require optimization and tight control
- Desire for autonomy



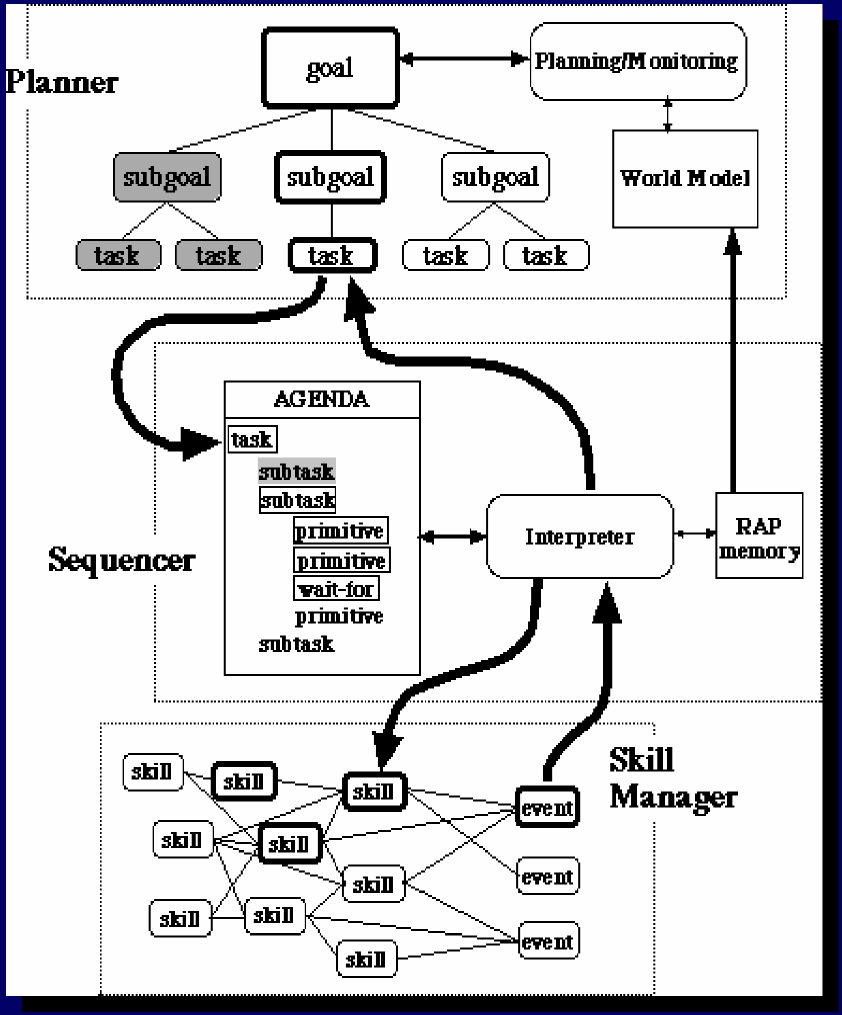
David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

Control Issues

- Advanced Life Support (ALS) systems are:
  - *Dynamic* – it is not sufficient only to find a single *a priori* setting
  - *Non-stationary* – presence of adaptive organisms such as humans, plants and bacteria as well as degradation requires adaptation
  - *Safety-sensitive* – crew depends on system for life support, verification and validation are important
  - *Coupled* – multiple heterogeneous systems (water, air, plants, people) all effect one another in obvious and subtle ways

Previous and Current Control Systems

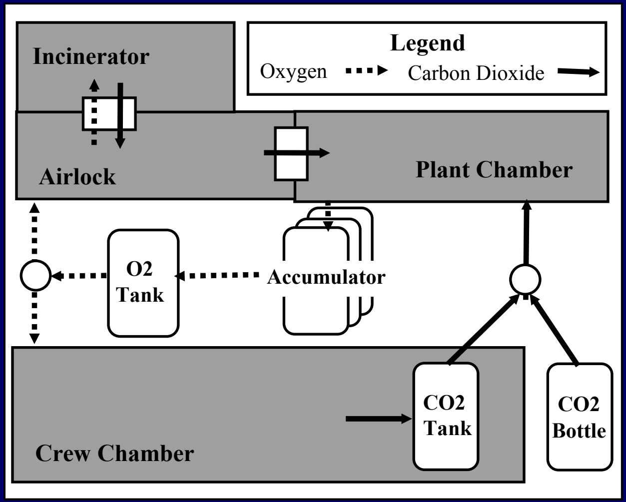
- Several experiments at JSC based on the 3T control architecture
- 3T
  - planning
  - sequencing
  - control



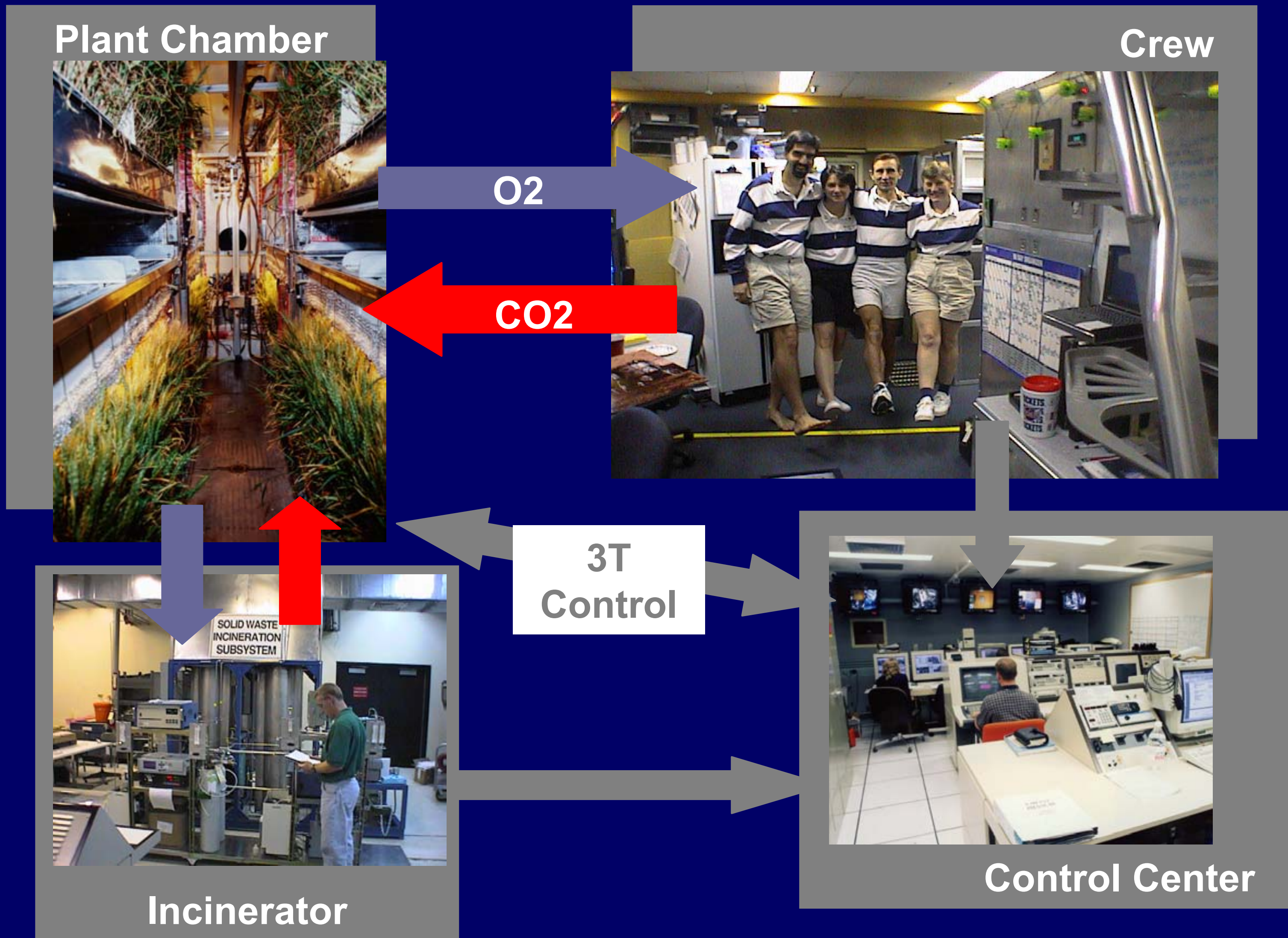
David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

Phase III Crewed Test

- Four crew members for 91 days in a closed chamber
- Wheat crop in another chamber
- 3T managed transfer of gases between the two chambers
- Operated reliably round-the-clock for 73 days (10/6/97-12/19/97)
- Typically ran without human supervision or intervention



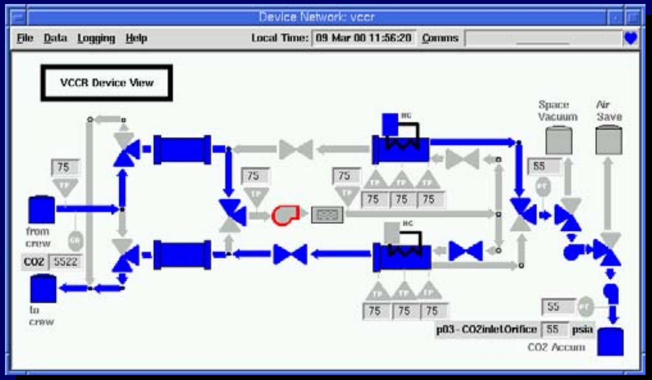
David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

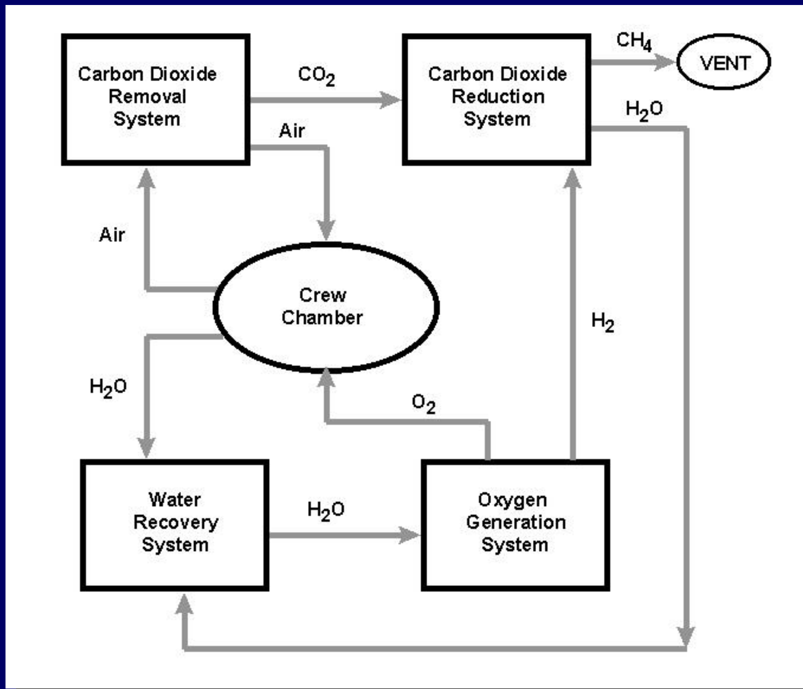


David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

Air Revitalization System

- Simulation of an ARS using a discrete event simulator
- 3T control (skills and RAPs) integrated with Livingstone MIR (from Ames)
- Planning currently being added

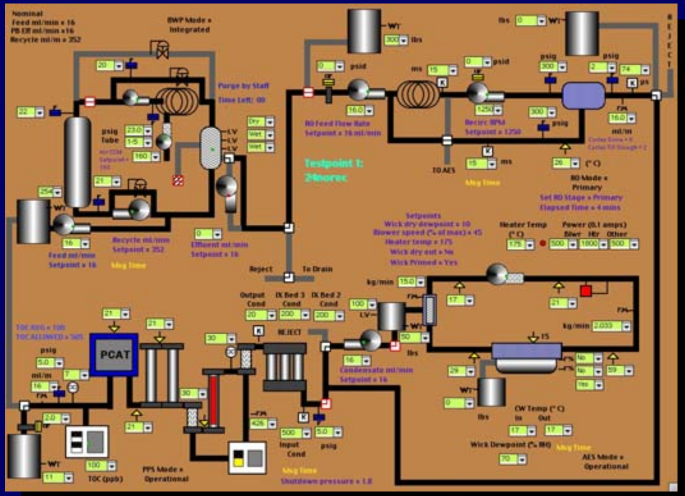




David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

Water Recovery System

- Four integrated subsystems:
  - Biological water processor
  - Reverse osmosis system
  - Air evaporation system
  - Post-processor
- 3T skills (over 75 separate skills)
- 3T RAPs
- ~200 sensors and actuators



David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

Lessons Learned

- Change in crew role from vigilance to supervision
  - System should let them know if there is something to be looked at
  - System should provide summaries of control actions
- Coupled systems
  - Full ALSS = WRS + ARS + plants + food production + climate control becomes planning/scheduling problem, not a control problem
  - AWRS had four subsystems itself – no planning

Lessons Learned 2

- Small changes to sensor calibration or the underlying biological/chemical processes requires expensive recoding of control procedures
- Changes to the desired operating regime (e.g., optimizing for a different resource) requires expensive recoding of control procedures
- Complex interactions are difficult to predict
- Adaptation of control code is required for long-duration, autonomous missions

Learning in ALS Systems

- Some of the control will be hand-coded and fixed
- Some portion will need to adapt as the system runs
- Many open research questions
  - On-line vs. off-line learning
  - Limits of experimentation with the real system
  - Fidelity of models and relationship to learning quality
  - Abstraction of state and action space (making system aware of hidden states)
  - Crew interaction with learning system (inspectability and instructability)

The Role of Learning

- Detecting signatures
  – Parsing real-world data stream to recognize events
- Refining models
  – Using feedback from actual system to adjust models
- Robust design
  – Searching through design criteria for optimal solution
- Learning/optimizing sequences
- Integrating with autonomous control
- Adaptive crew interfaces
- Control system design methodology
  – Using learning algorithms to find important variables and interactions
- Optimizing resource allocation

ALS State Space

- Potential state space is enormous and hybrid (i.e., mix of discrete and continuous) so we need to abstract
- Possible abstractions are
  - Current levels of consumables (air, water, food)
  - Quality of air and water and health of plants
  - Flow paths for water and air through the system
  - Current energy allocations to subsystems
  - The current phase of operation
  - Crew health/happiness
  - Temperatures and other environmental measures

ALS Action Space

- Potential action space is large and hybrid
- Combination of physical actions to produce abstract actions
  - Allocation of energy amongst subsystems
  - Use of consumable stores
  - Crew activity
  - Routing of air/water flows
  - Planting/harvesting of crops (when and which)
  - Adjusting crop light levels
  - Adjusting climate controls
  - Venting of gases to the outside atmosphere

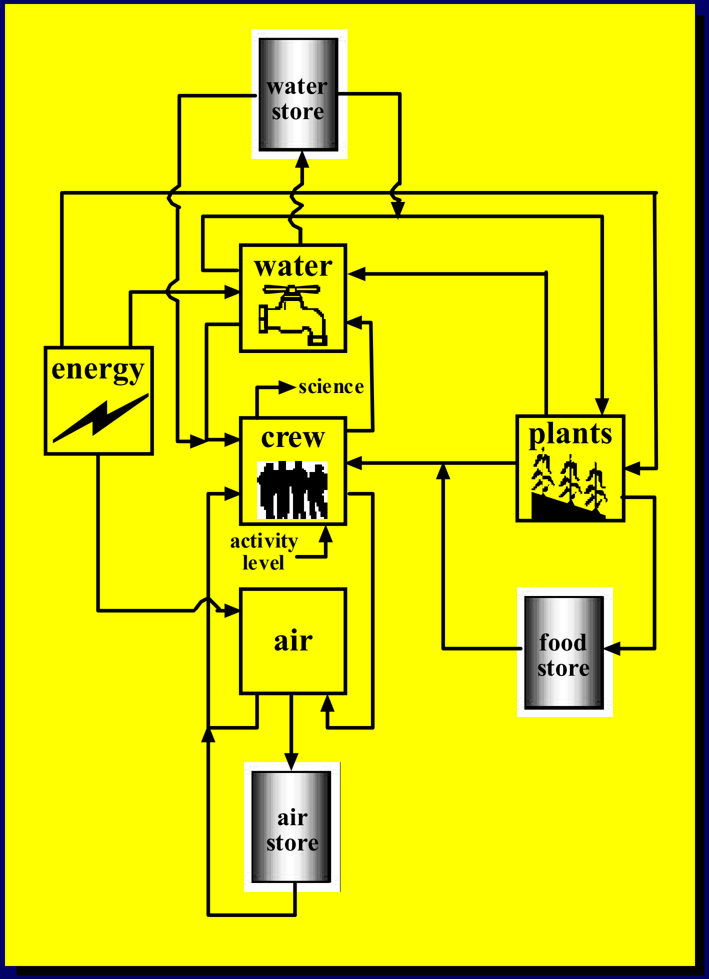
ALS Rewards and Feedback

- Final or end state rewards
  - Duration of mission with different controllers
  - Total crew productivity over mission duration
  - Total amount of air, water, food or energy available in system or stores
- Progress measures
  - Quality of air and water and health of plants
  - Plant growth rates and plant food output
  - Climate feedback (keeping climate parameters within boundaries)
  - Health/satisfaction of crew

Experiments with a Simplified Simulation

- Goal was to determine if simple ML techniques worked on simple ALS simulation. Then scale up each accordingly
- Simple, deterministic simulation
  - Air, water, plants and crew modules
  - Simulation time in ‘ticks’ (one hour)
  - Single tick runs each process once
  - Each process takes energy and dirty resources and produces a resource (clean air, clean water or food)
  - Crew consumes clean resources and produces dirty
  - Stores for food, water and air

The Simulation



David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

Simulation Resource Rates

- Low activity level
  - dirty water (W_d) = 0.95 clean water (W_c)
  - dirty air (A_d) = 0.95 clean air (A_c)
  - Science = (food + W_c + A_c) * 0.30
- Medium activity level
  - dirty water (W_d) = 0.85 clean water (W_c)
  - dirty air (A_d) = 0.85 clean air (A_c)
  - Science = (food + W_c + A_c)* 0.60
- High activity level
  - dirty water (W_d) = 0.75 clean water (W_c)
  - dirty air (A_d) = 0.95 clean air (A_c)
  - Science = (food + W_c + A_c)* 0.90

David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

ML Experiments

- Goal
  - Find control policy that maximizes either mission length (i.e., ticks), total science or combination of both
- Assumptions
  - Markov process
- Experimented with
  - Reinforcement learning (search in value space)
  - Genetic algorithms (search in policy space)
- Developed a novel genetic algorithm approach

David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

Reinforcement Learning

- Q learning

$$Q(s,a) = (1-\alpha)Q(s,a) + \alpha[r + \max_{a' \in A} Q(s',a')]$$

- Action space
  - Energy levels (increase or decrease), activity level (increase or decrease), use of stores
- State space
  - Energy levels, activity level
- Reward
  - Change in science at each tick (when maximizing science)
  - Increment of one for each tick (when maximizing duration)
  - End of mission (negative reward)

Results

Samples	science	ticks
502	150.448308	13
1018	318.30795	20
1525	179.515967	30
2037	318.30795	20
2548	318.30795	20
3059	322.06935	20
3577	318.30795	20
4078	318.30795	20
4594	318.30795	20
5111	322.06935	20
5629	322.06935	20
6009	322.06935	20

Reward for science

Samples	ticks	% Random
1011	28	90
2014	25	80
3035	28	70
4037	28	60
5070	28	50
6082	28	40
7107	28	30
8217	38	20
9276	38	20
10080	38	20

Reward for mission duration

Genetic Algorithms

- Bit string represents actions (or inputs) into the system

10 | 10 | 10 | 11 | 0 | 1 | 0

water | air | plants | crew activity level | water | air | food

energy allocation | crew activity level | stores

- Simulation used to evaluate strings

David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

Results

Exp	Ticks	Science
1	31	147.7
2	31	147.5
3	31	49.2
4	31	49.2
5	31	49.2
6	31	51.9
7	31	52
8	31	147.5
9	31	140.9
10	31	147.5

evaluation = mission duration
i.e., ticks ^ 5

Exp	Science	Ticks
1	183.9	26
2	184	26
3	183.9	26
4	184	26
5	184	26
6	184	26
7	183.8	26
8	183.8	26
9	184	26
10	184	26

evaluation = science cubed

What’s Wrong?

- We should be doing better, even simple control strategies achieve better results
- Hypothesis
  - Our abstraction of the state and action space is missing some key aspect of the problem

Multi-step Genetic Algorithms

- Bit string encodes a multi-step "plan"

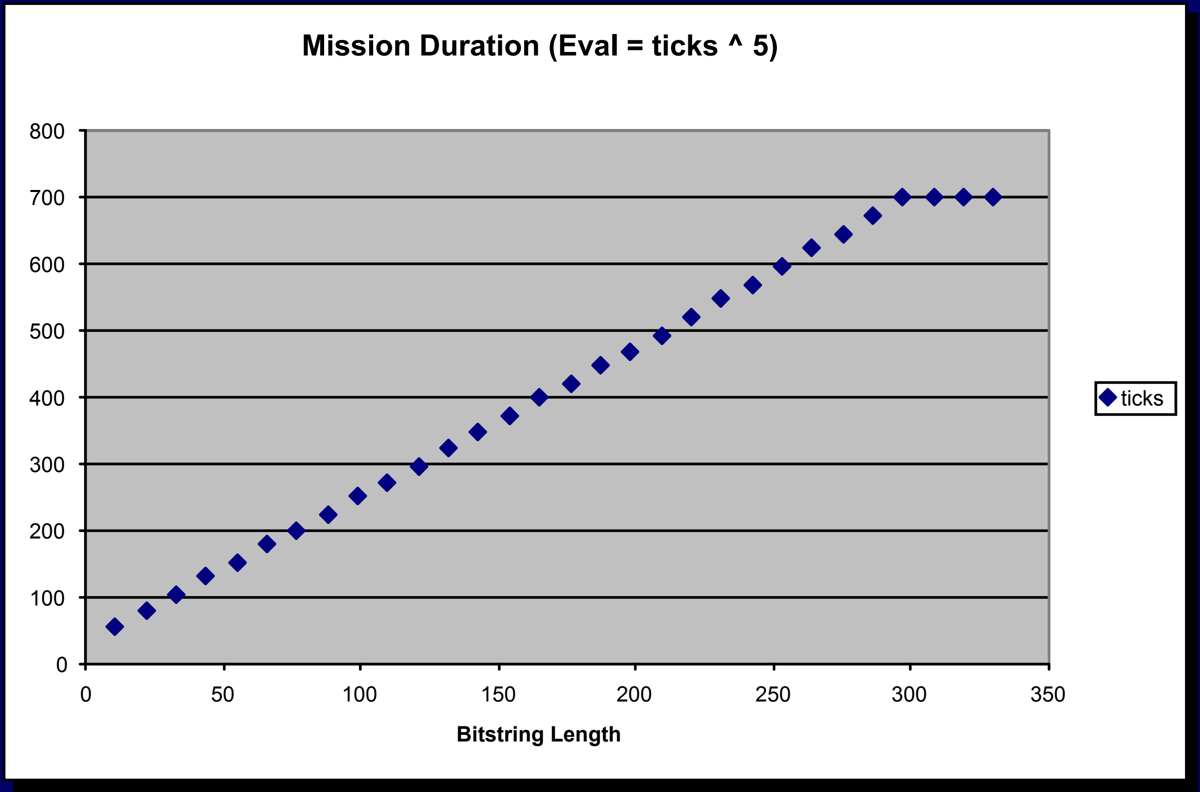
10101011010	10100011011	00101111010	10111010110
policy for first tick	policy for 2nd tick	policy for 3rd tick	policy for 4th tick

- Bit strings are *n* times 11 in length where *n* is plan length before repeating

David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

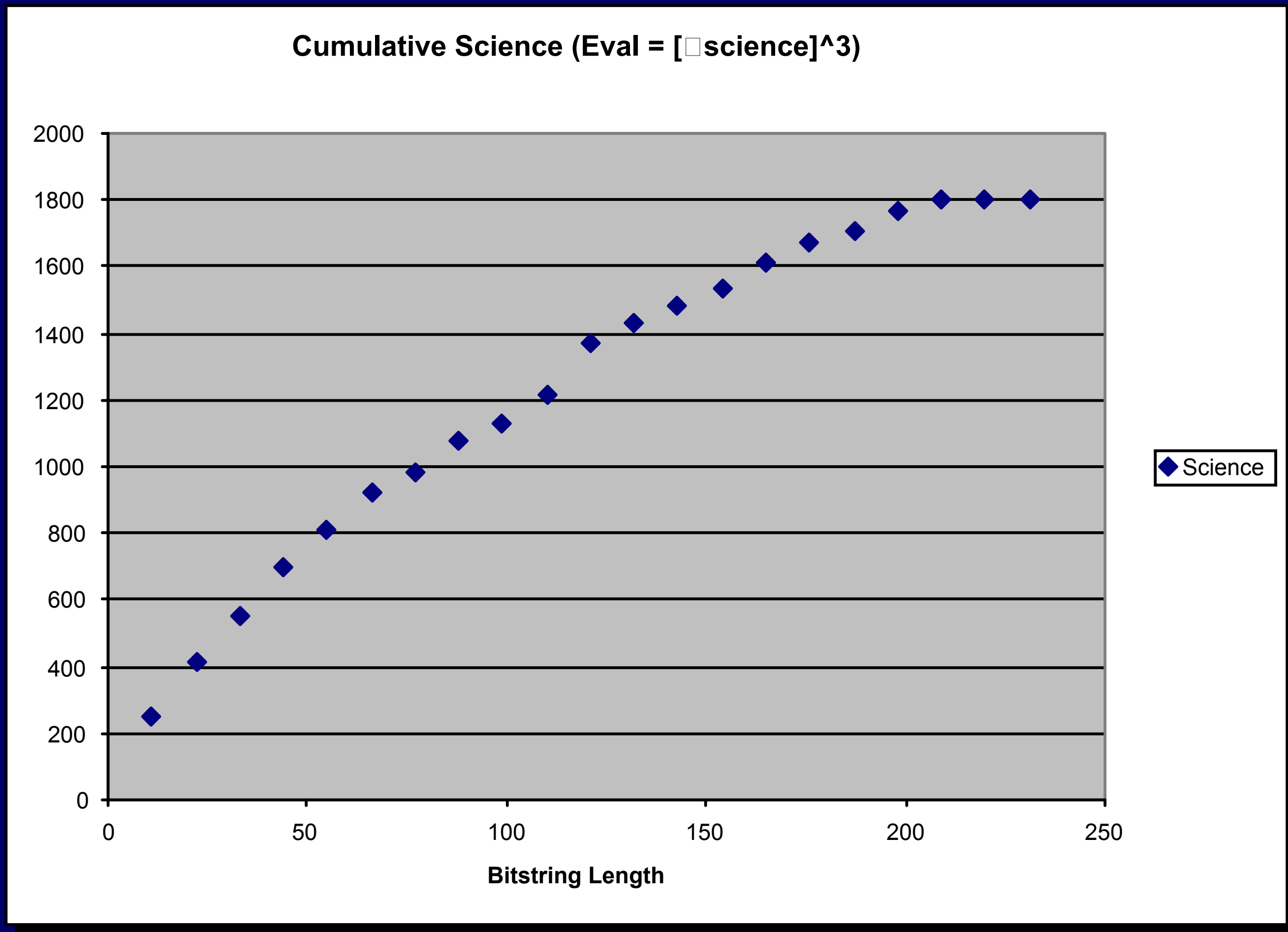
Results

- Results were stunning



Compare to 31 ticks max using “normal” GA

Results (cont.)

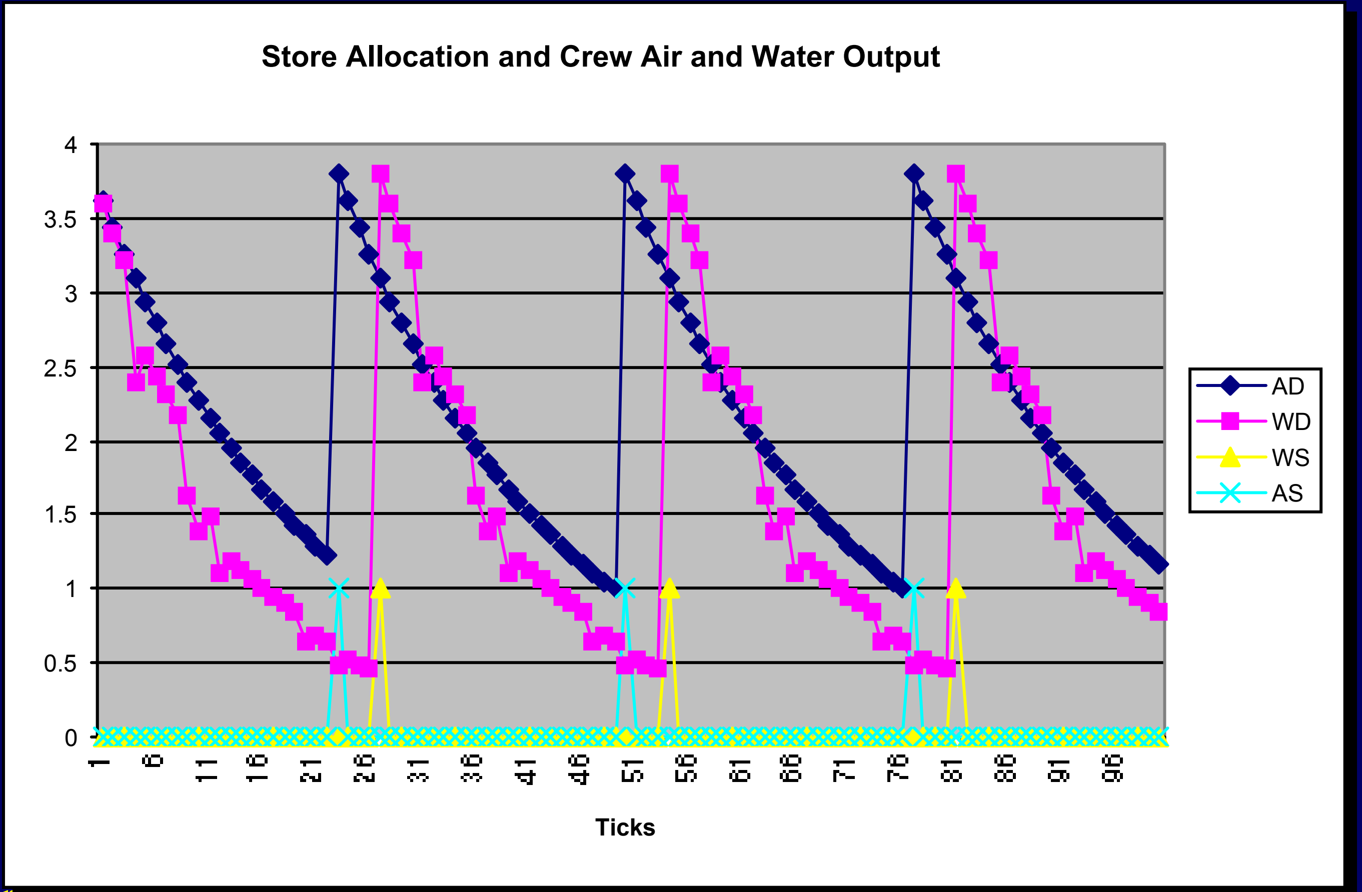


David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

Compare to 184 total science max using "normal" GA

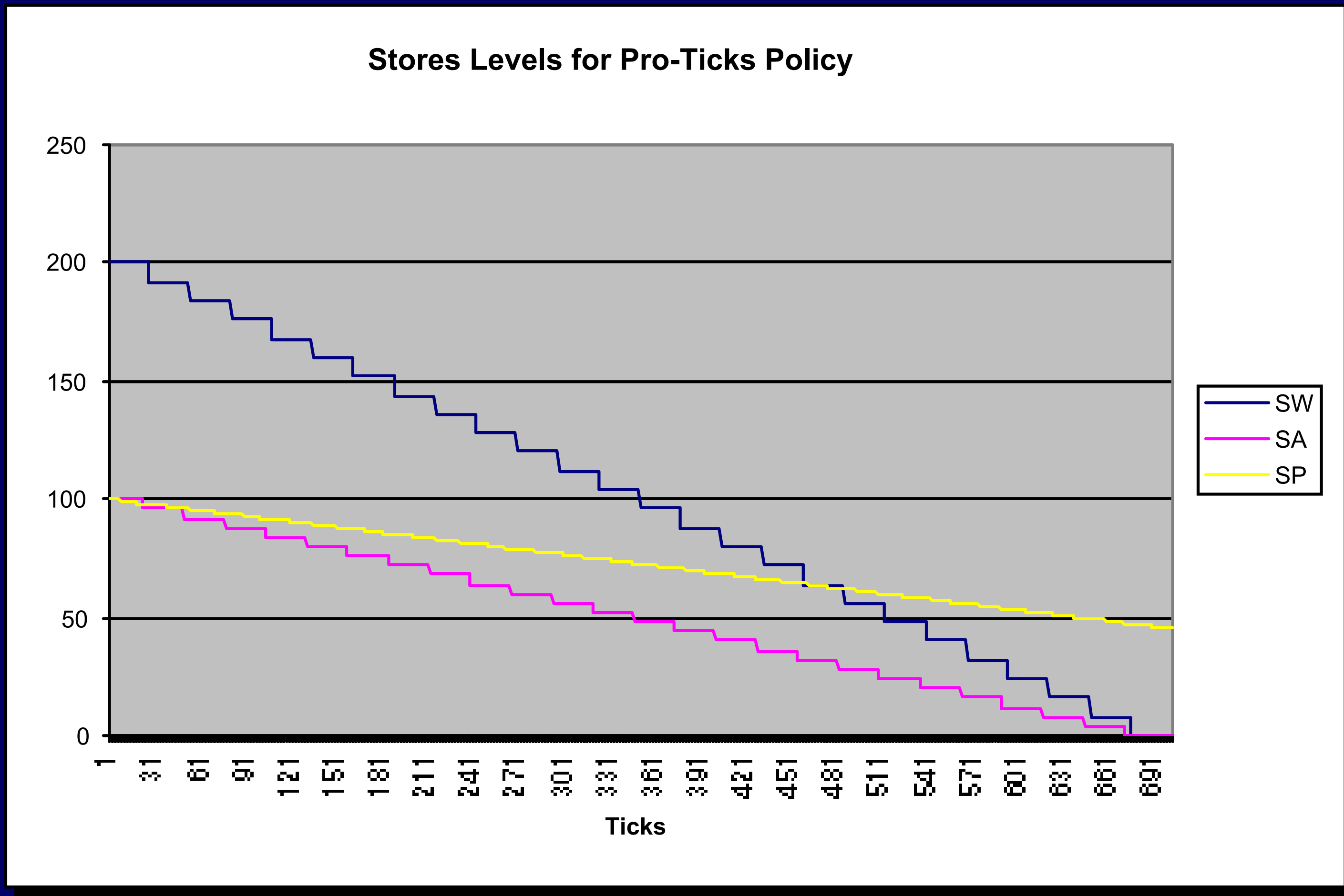
What’s Happening?

- GA is learning a “pulsing” strategy



David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

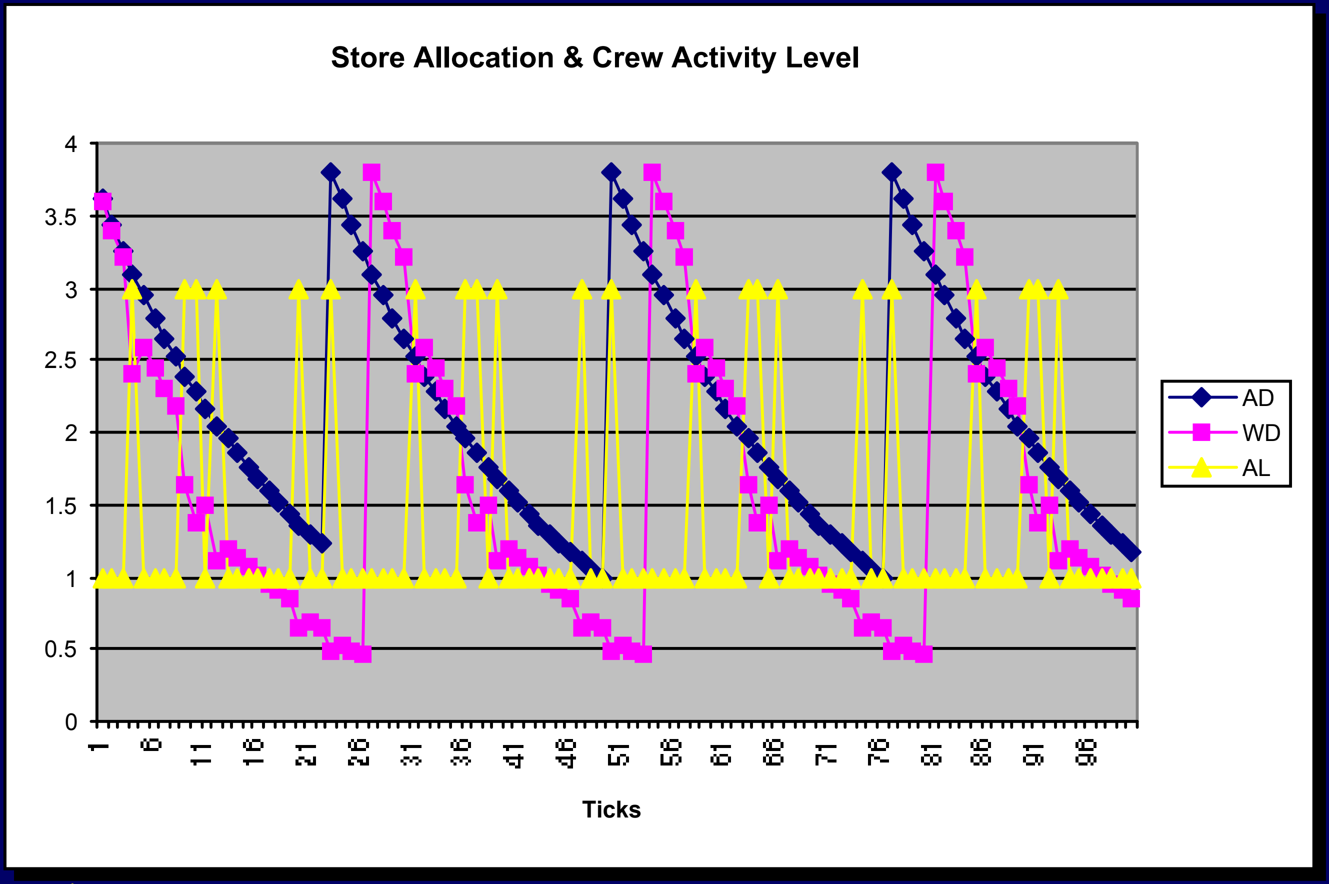
Eking out an Existence



David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

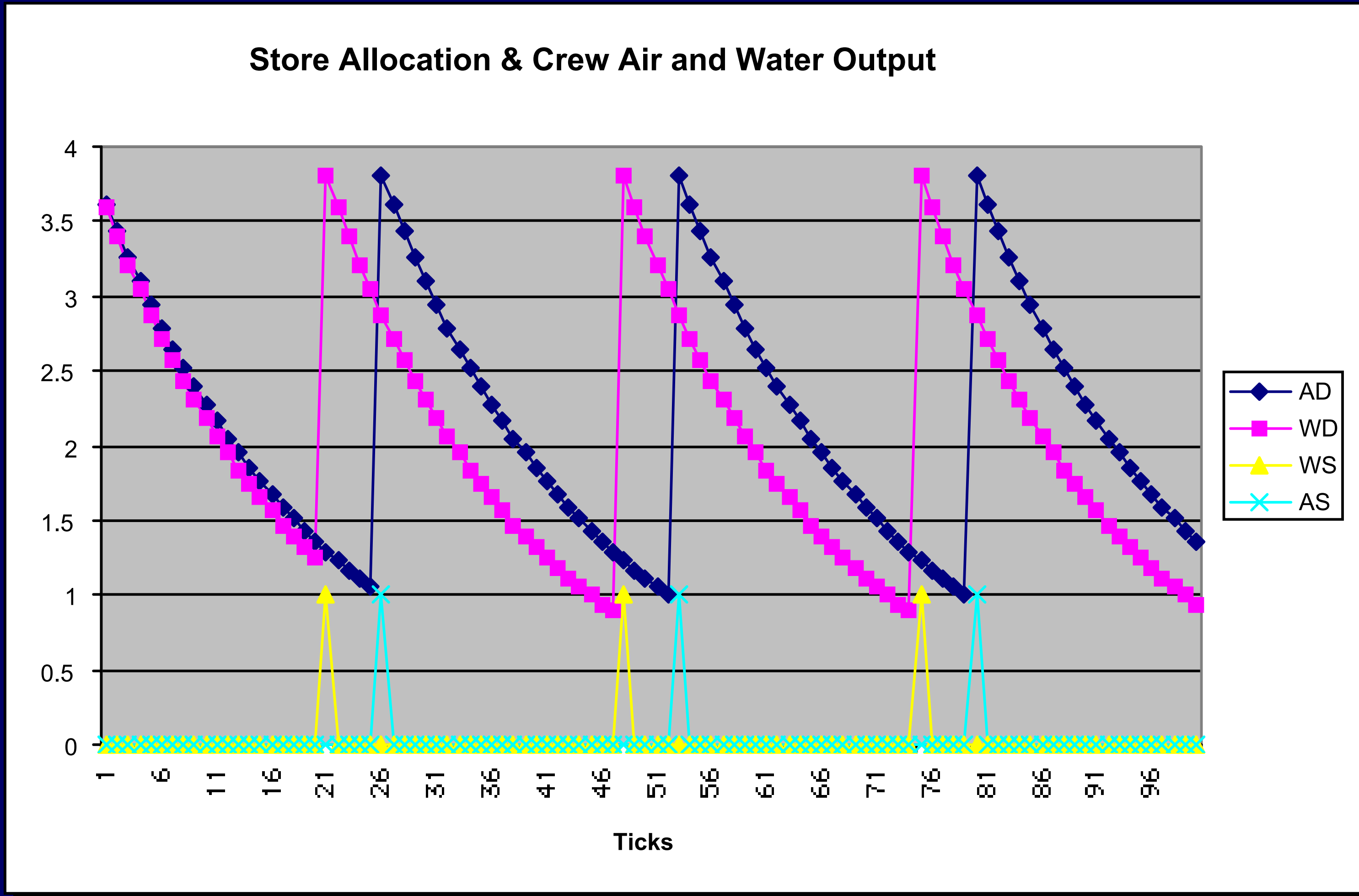
Informing a GA

- How to use human intuition to help a GA along



David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

Results



Activity level constrained to 1

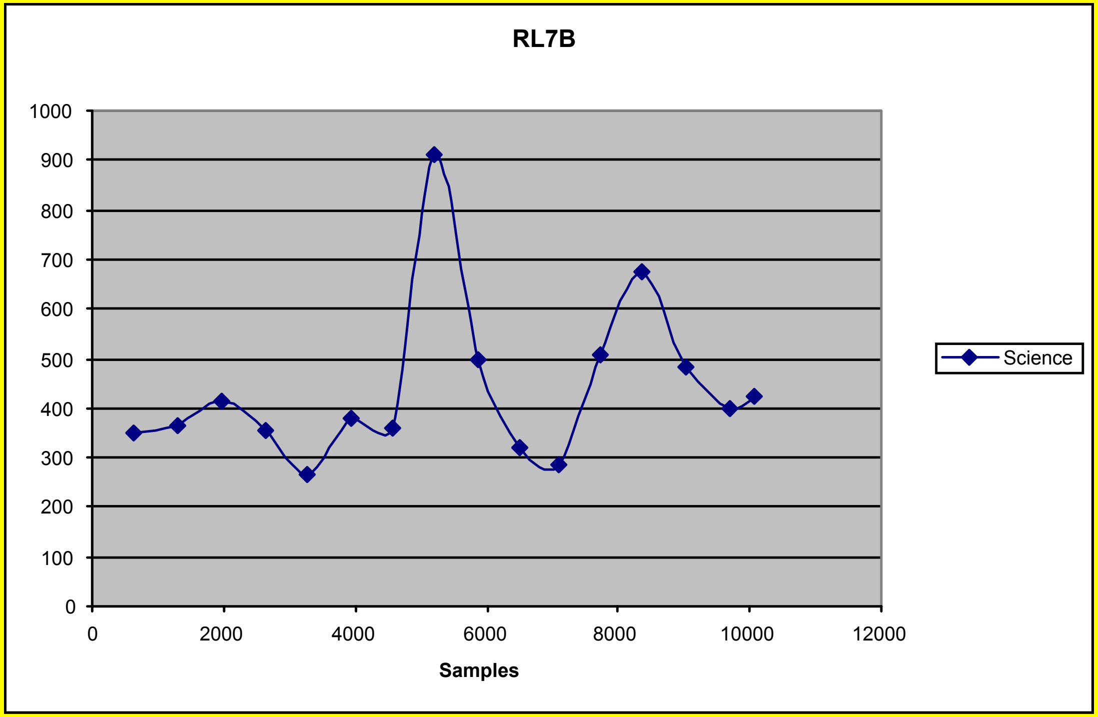
David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

Revisiting Q-Learning

- Our initial state space was impoverished and open loop (but GA does not need an explicitly designed state space and is also open loop)
- New state space
  - Water and air outputs from the crew (discretized to 0-2)
  - Air, water, food store status (0 or 1)
  - Ticks since last action for air and water stores
- New action space
  - To use or not use water, air or food store

Results still not as good as multi-step GA, but much better than original RL

New RL finds same "pulsing" strategy



Samples	Science	Ticks
647	350.166058	67
1295	362.099302	67
1951	411.486381	68
2622	355.082955	61
3267	264.029052	45
3925	379.153828	68
4554	361.391647	61
5213	909.616541	176
5865	496.283236	92
6500	319.51492	61
7108	284.349158	49
7718	505.187961	92
8361	672.872989	123
9015	482.361459	87
9681	398.32097	74
10068	423.490573	77

David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

Other Experiments

- Optimizing both mission duration and science output at the same time
- Optimizing in the face of finite energy
- Optimizing in the face of finite time
- Feeding the results of GA back into RL

Open Questions

- GA is open-loop and has no guarantees, however, it does not require explicitly defining state
- Can we use results of GA to design a state space that will let RL do better?
- Can we use results of GA to hand-code a better controller?
- Research question: How can different learning algorithms be used to compliment each other?
- How can human intuition be used to guide machine learning algorithms in their search?

On-going Work

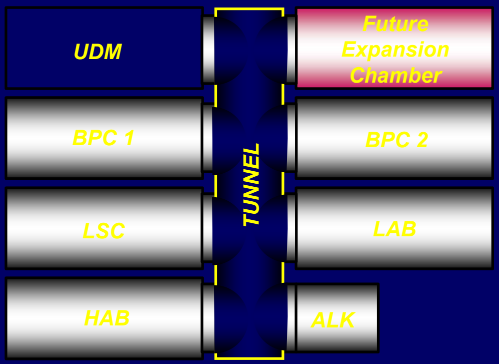
- Developing a more challenging simulation
  - Stochastic processes
  - Cycles (plants, crew)
  - Model subsystems within water, air and plants
  - Portable (Java) and available to all
- Objective
  - Make simulation more realistic
  - Re-test machine learning techniques and see what changes are necessary

Future Work

- Begin working on other machine learning application areas
  - Sequence learning
    - Learning contexts as well as sequences
  - Integration with control systems
    - How good does initial control have to be for on-line learning to work?
    - How does control system decide when to devote resources to learning and when to use new knowledge?
- Investigate other ML techniques (memory-based, Samuel)
- Continue to explore theoretical issues of abstraction and model fidelity requirements
- Issue challenge to AI research community and make simulation available to all
- Begin applying techniques to real-world ALS testbeds

Integrated Control of ALSS

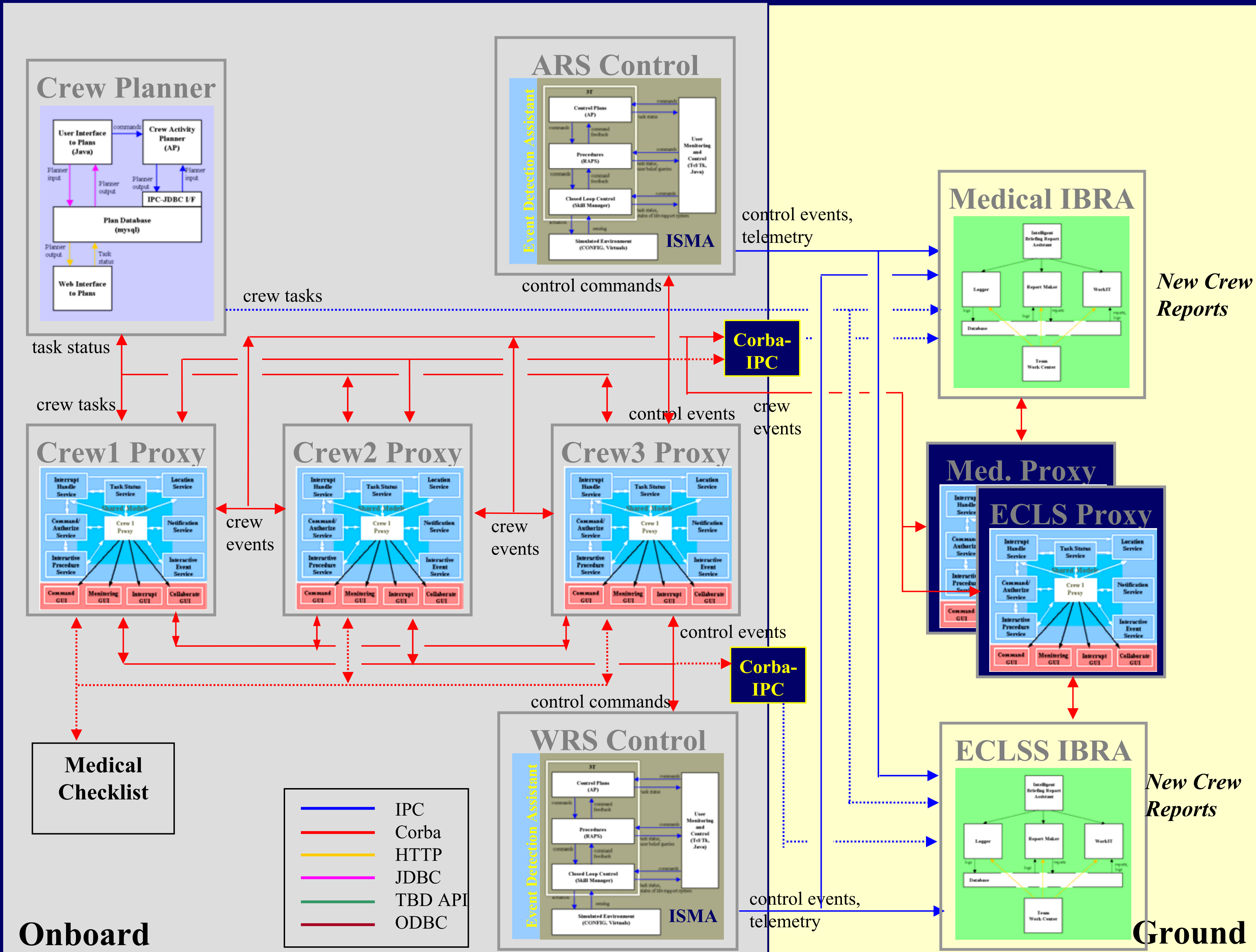
- Distributed, integrated monitoring and control architecture
  – Dynamically reconfigurable
  – Integration of data from multiple sources
  – Distributed problem solving
- Software methodology for developing, debugging, verifying and validating such an architecture
  – Modeling tools and languages
  – Automated code generation from high-level specifications
- Coupled biological processes whose dynamics change over time
  – Dealing with randomness and implementing adaptation
- Hybrid discrete/continuous processes
  – Processes distributed over space and time
  – Processes that have widely varying time constants
- Failure modes that accumulate over time
- Optimization of finite consumables
  – Adaptation of standard operating procedures
  – Automated task planning and scheduling
  – Optimizing time, space and consumables



David Kortenkamp
NASA Johnson Space Center/Metrica Inc.

Integrated Control of ALSS (cont.)

- Integrating of human expertise (ground and crew) with autonomous control
  - Mechanisms and support for humans to adjust the level of autonomy and/or change the distribution of roles and responsibilities between autonomous control and humans
  - Mechanisms for the allocation of and control of initiative among humans and the autonomous control system
  - Circumstances and methods by which the autonomous control system notifies humans of environmental events (nominal and off-nominal) and accepts task inputs from human


Crew Planner
User Interface to Plans (Java)
Crew Activity Planner (AP)
IPC-JDBC I/F
Plan Database (mysql)
Web Interface to Plans
ARS Control
Event Detection Assistant
ISMA
control commands
crew tasks
task status
crew tasks
control events, telemetry
Medical IBRA
New Crew Reports
Corba-IPC
control events
crew events
Crew1 Proxy
Crew2 Proxy
Crew3 Proxy
crew events
crew events
Med. Proxy
ECLS Proxy
control events
Corba-IPC
control commands
WRS Control
Event Detection Assistant
ISMA
ECLSS IBRA
New Crew Reports
control events, telemetry
Medical Checklist
IPC
Corba
HTTP
JDBC
TBD API
ODBC
Onboard
Ground

ALSS Autonomy Roadmap

Planning/Scheduling	Simple task planning for single subsystems	Planning for different time scales	Mixed-initiative and crew activity planning	Crop and menu planning	Continuous planning and replanning
Executive	Single system procedures	Probabilistic reasoning about task context	Distributed, cooperating executives	Reasoning about procedure execution	Procedure synthesis
Machine Learning	Parameter tuning	Learning/refining models from system data	Learning cross-system optimal control policies	Optimizing control in changing environments	Continuous learning
Model-based Reasoning	Multi-step reconfiguration	Hybrid discrete/ continuous models	Hierarchies of models for reasoning across subsystems	Modeling and reasoning about software procedures	Procedure synthesis from models
Sensor Interpretation	Sensor fusion	Automatic event recognition	Automatic sensor calibration	Interpretation of sensor nets	Vision for crop inspection and crew tracking
Distributed Control	System architecture	Communication protocols and APIs for distributed components	FDIR on control system components	Dense networks of distributed sensors	Automated recovery from major control system failures
Human Interaction	Natural language discourse with control system	Situation views of control system status	Mixed-initiative planning interfaces	Mobile computing for control system	Crew tracking and plan recognition
Robotics	Autonomous control of Traybot	Planning and control of simulated robots	Shared control of EVA rovers and maintenance robots	Plant chamber automation for food processing	Test of BIO-Plex IVA maintenance and inspection robot
Intelligent Data Understanding	Data models for storing data in database	Use of stored data to automatically refine models and simulations	Automated inventory control system	Automatic identification of significant events	Sophisticated analysis of trends and events
Tests	Simple ground demonstration	Ground demo with crew	Station flight demonstration	Long-term station deployment	Beyond LEO deployment

David Kortenkamp
NASA Johnson Space Center/Metrica Inc.