Characterizing the Performance of Applied Intelligent Agents in Soar

Randolph M. Jones, PhD

Steve Furtwangler

Mike van Lent, PhD

Soar Workshop 2011

Definitions

- Research agents
  - Agents built primarily for scientific goals (as opposed to application goals)
    - To explore cognitive mechanisms and capabilities
    - Explicitly to evaluate the architecture
- Applied agents
  - Agents developed for DoD projects, for which:
    - Evaluation of Soar was not a goal in the agent design
    - Primary design and implementation goal was to meet application-specific reasoning and behavior generation requirements
    - Efficiency optimization efforts were of secondary emphasis, as long as the systems met the project requirements
- Acceptable performance
  - Execution time and memory requirements should not increase over time
  - Decision-cycle time for human-level reactivity is 50 msec (Newell, 1990)

SOARTECH

Hypotheses

- The Soar performance results demonstrated by other studies will also carry over to applied agent systems
  - H1: Over a long-running task, memory usage will initially increase monotonically then reach an asymptote
  - H2: Over a long-running task, execution time will initially increase monotonically then reach an asymptote
  - H3: Decision-cycle time will be well within the 50 msec theoretical limit for human-like reaction
- Because applied agents are not primarily optimized for performance, there are ready opportunities to improve them
  - H4: Simple adjustments will improve performance, with diminishing returns for iterative improvements

Addition empirical question

- Using 50 msec reaction time as a limit, what is the order of magnitude for the number of applied intelligent agents that can be expected to run within acceptable performance constraints on typical modern computer hardware?

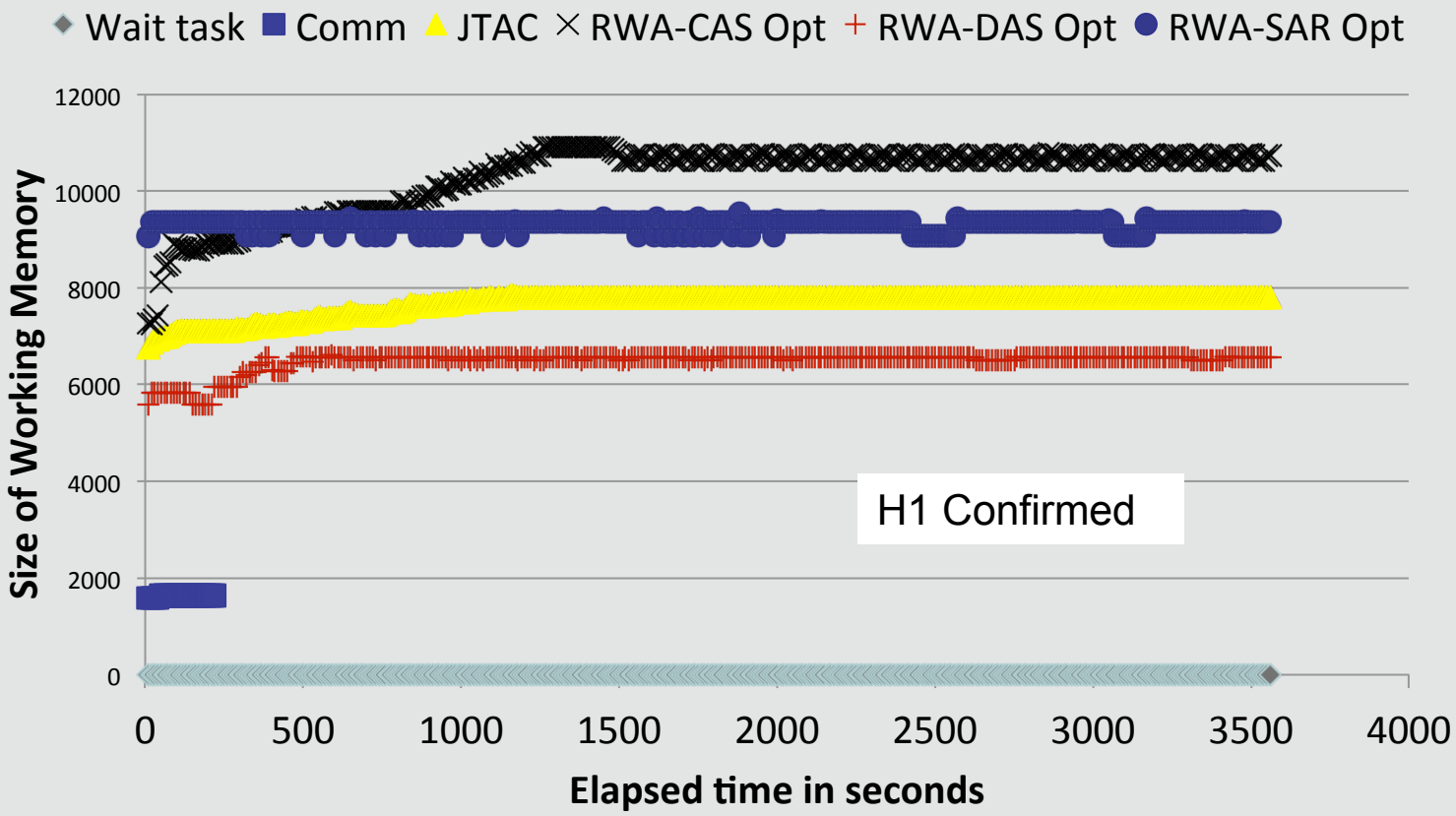
Soar Agents for Evaluation

Agent	# of productions	Avg. WM size
Research Systems (Laird et al., 2011)		
Simple robot	22	~250
Complex robot	530	~3000
Applied Systems (this study)		
Comm	224	~1600
JTAC	274	~7600
RWA-CAS	2045	~10500
RWA-DAS	2045	~6500
RWA-SAR	2045	~9000

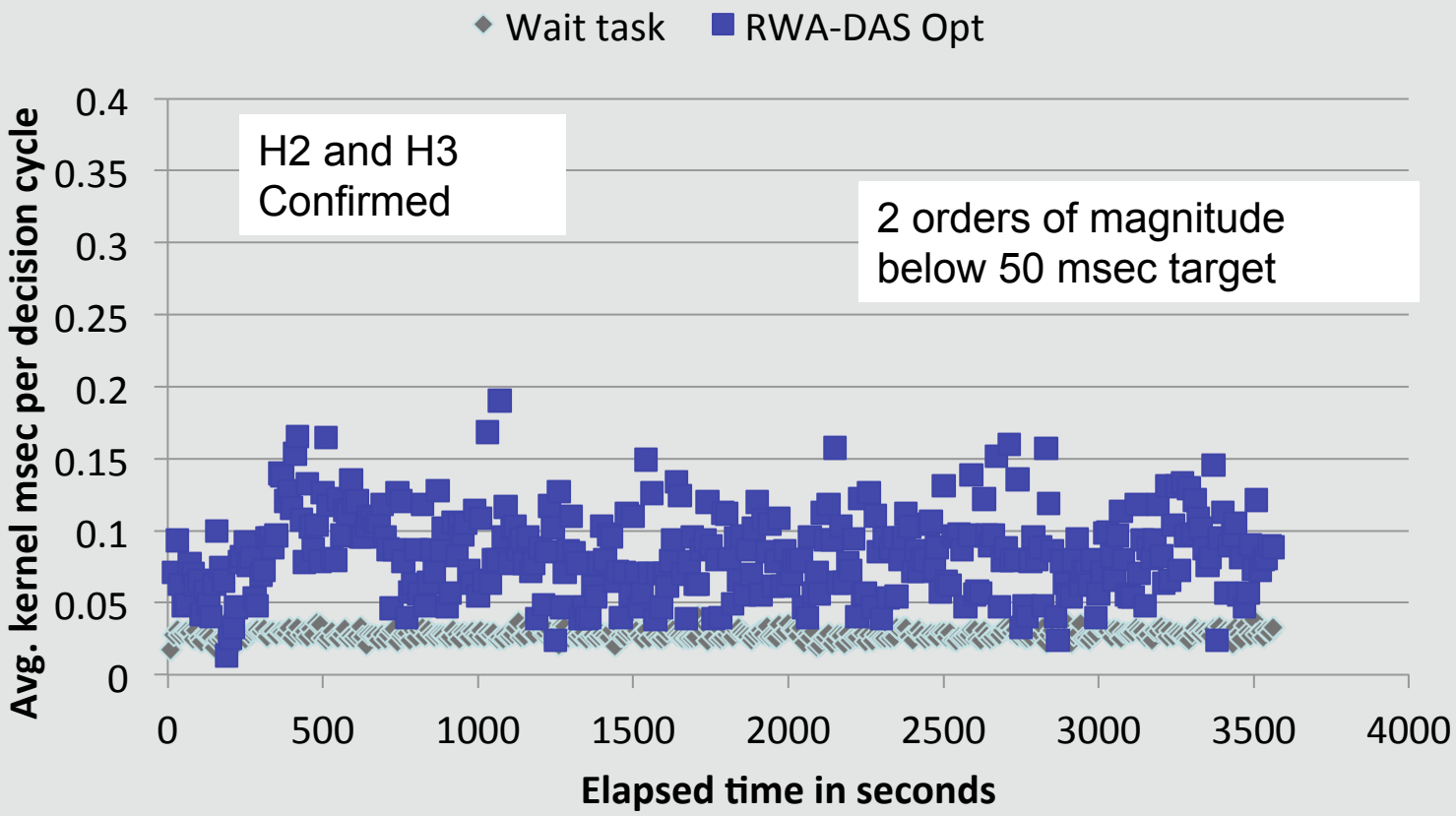
Agent environment for experimentation: SimJr

- Developed and maintained at SoarTech
- Features:
  - Modular and extensible Java implementation
  - Low-fidelity models of fixed- and rotary-wing aircraft
  - Support for high-fidelity/intelligent model control
  - Exploits Java's integration tools and environments
  - Fully open-source
- Available from:
  - http://code.google.com/p/simjr/

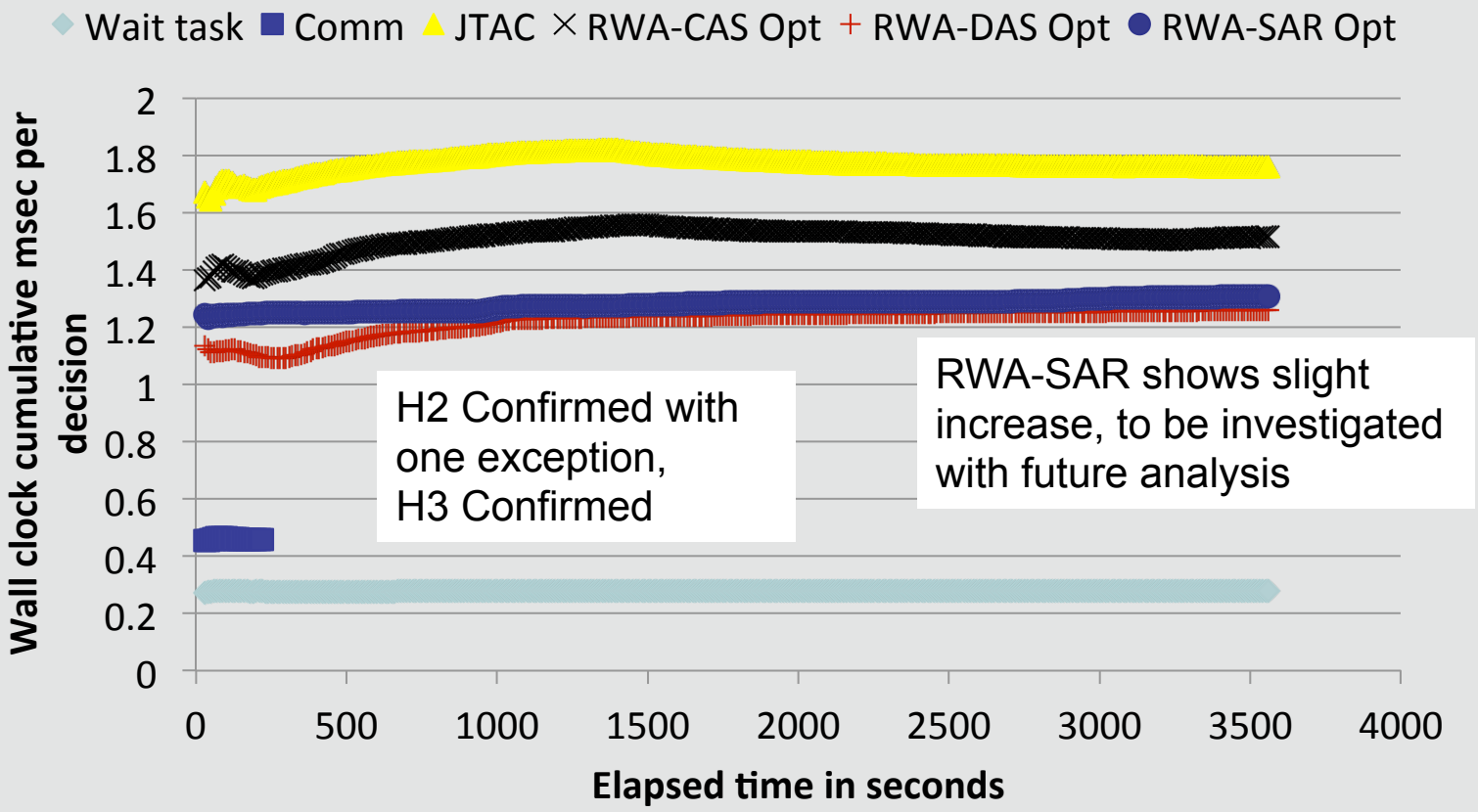
Result: Working memory size does not increase over time



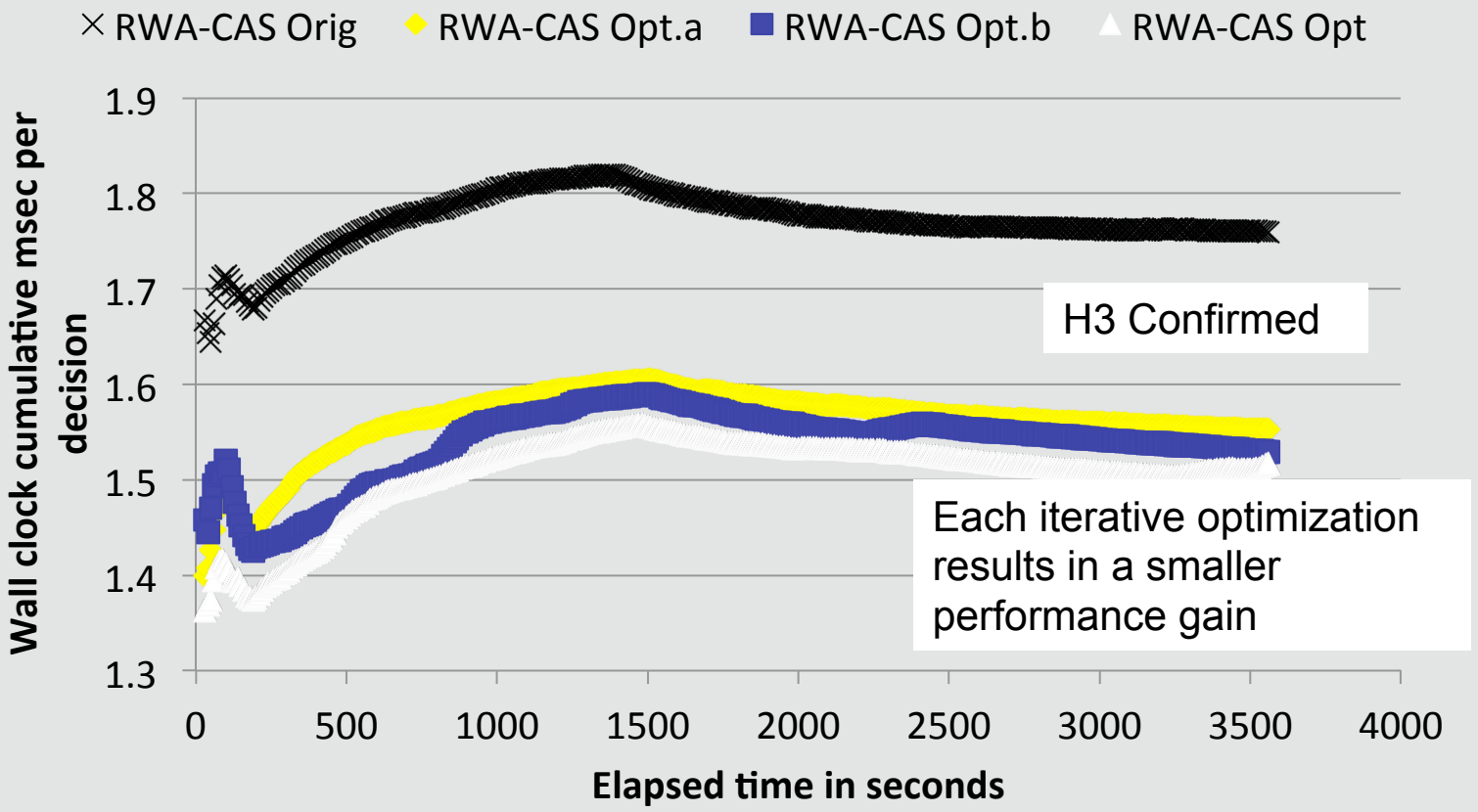
Result: Kernel time/decision is very low and does not increase over time



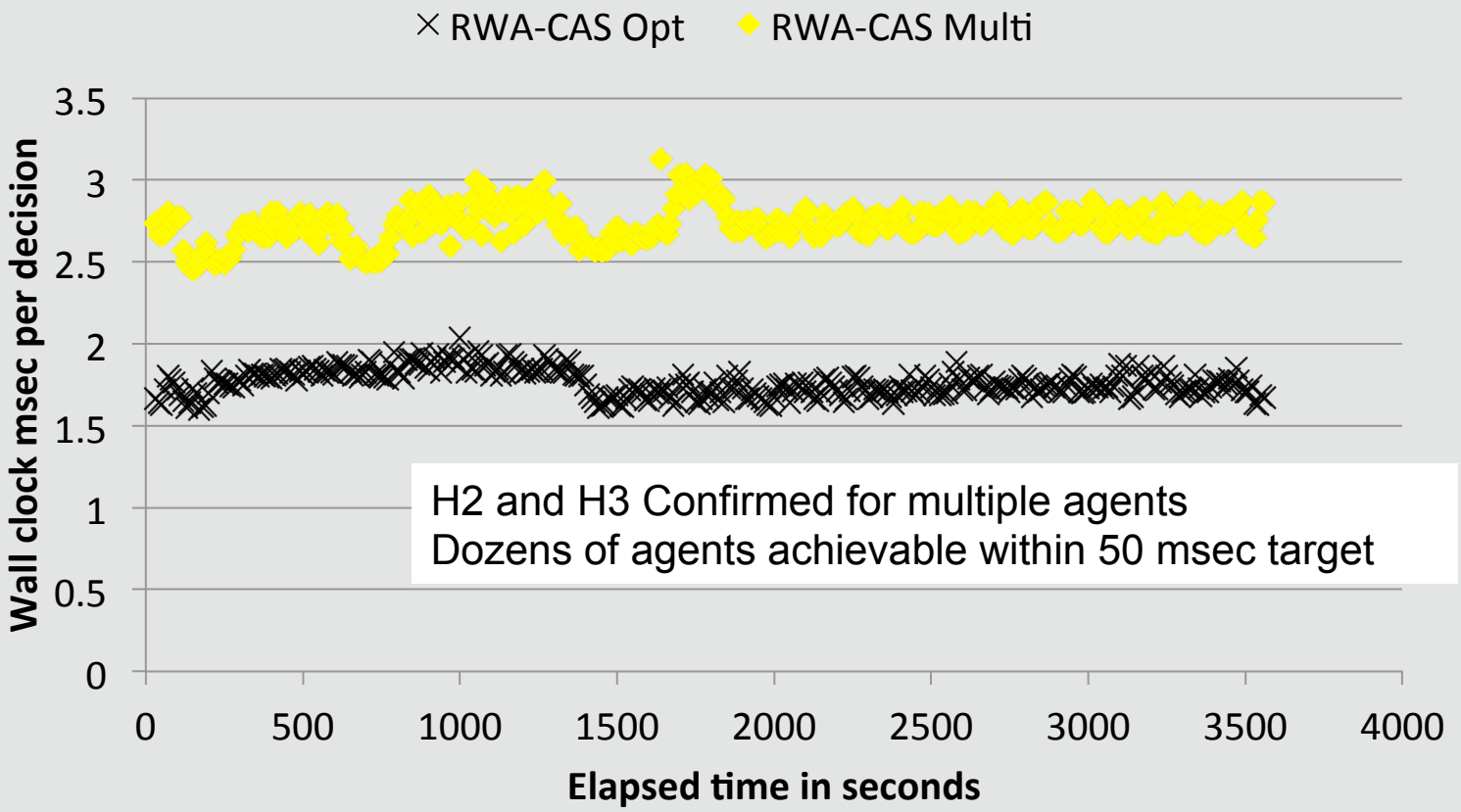
Result: Wall clock time/agent-decision is very low and does not increase over time



Result: Simple optimizations improve execution time even further



Result: Multiple agents on a machine increase execution time slightly (cause to be determined)



Conclusions

- Soar performance results demonstrated by research agents also carry over to applied agent systems of varying complexity
- Performance of applied Soar agents does not degrade over the course of long-running tasks
- There are opportunities to improve the performance of applied agents that were not designed with optimized performance as their primary goal
- The number of applied agents of fairly high complexity that can be expected to run within acceptable performance constraints on typical, modern computer hardware is in the dozens
- Future work:
  - Further experiments to provide full explanations for some cases of gradual performance slowing and increased overhead in multi-agent scenarios