Cognitive Constraint Modeling: An Alternative to Traditional Architectures

Andrew Howes[1] Richard Lewis[2] Alonso Vera[3]

[1]School of Informatics
University of Manchester

[2]Department of Psychology
University of Michigan

[3]HCI Group
NASA Ames and Carnegie Mellon

May 25, 2006

Key Claims

1. Human Task Performance can be predicted by **formally reasoning about the implications of a theory** rather than running a simulation.
2. A theory of cognitive architecture **explains** empirically observed asymptotic bounds on performance if there is **substantial correspondence between the asymptote and the optimal performance implied by the theory.**
3. The ability to automatically derive optimal predictions from cognitive theory has **significant theoretical and applied benefits**.

How Architectures Make Predictions

ARCHITECTURE + KNOWLEDGE (STRATEGY) = BEHAVIOR

A Conundrum for Cognitive Theory

Complete cognitive theories must take the form architectures that admit of arbitrary knowledge/strategic variation

BUT: knowledge, strategy can become theoretical degrees of freedom in modeling data

- Explanation may reside primarily in strategy, not architecture
- Strategy may have been selected to fit the data at hand

A Conundrum for Cognitive Theory

Complete cognitive theories must take the form architectures that admit of arbitrary knowledge/strategic variation

BUT: knowledge, strategy can become theoretical degrees of freedom in modeling data

- Explanation may reside primarily in strategy, not architecture
- Strategy may have been selected to fit the data at hand
- *(But that never happens, right?)*

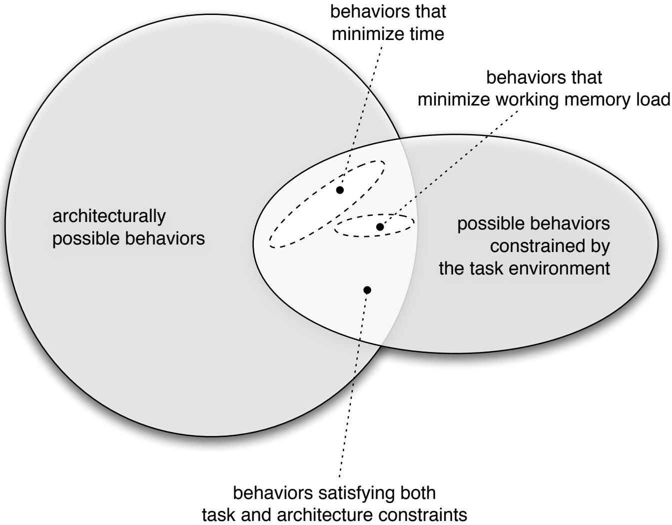
Two Possible Solutions

1. Focus on **“immediate behavior”** (Newell 1990)
   - Behavior < 1 s
   - Problem: Even < 1 s behavior shows surprising amount of strategic modulation (Meyer & Kieras, 1997)
2. Theory of **learning/instruction taking**
   - “Close the loop”, so strategy not under theorist’s control
   - Problem: complexity; testing many aspects of theory simultaneously

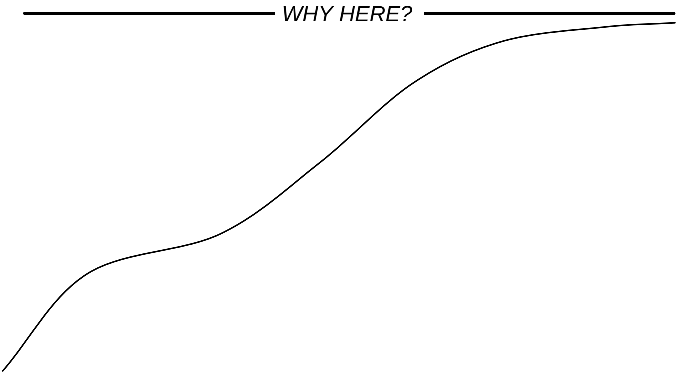
Constraint Analysis Overview

- Adaptive behavior is bounded by **Objective + Environment + Knowledge + Architecture** (Simon 1992)
- **Constraint satisfaction techniques can be used to calculate the optimal behavior given a set of heterogeneous constraints plus an objective.**
- In short, combining **Formal Rational Analysis** with **Bounded Rationality**

Constraint Analysis Overview

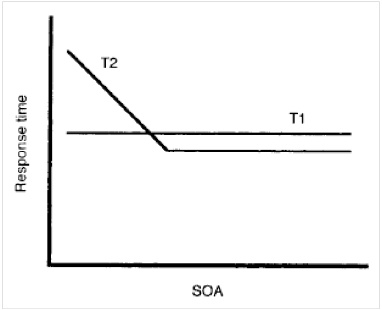


Explaining the Bounds on Adaptation



Typical PRP (psychological refractory period) Experiment

- Choice response to a tone (T1) and a pattern (T2).
- Give priority to the tone response.
- Tone presented first, pattern stimulus is presented after an SOA.



- According to Meyer and Kieras, elevated RT2 is because participants ensure T2 response is after T1 response
- They called this *Strategic Response Deferment* (SRD).

Simple Dual-Tasking PRP Study

Ruthruff et al., 2003 report a PRP experiment with:

- Single participant.
- Unordered responses.

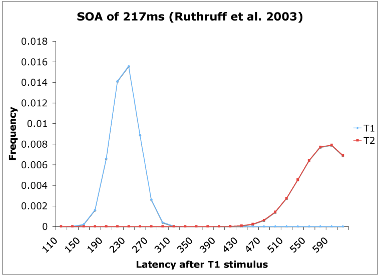
Simple Dual-Tasking PRP Study

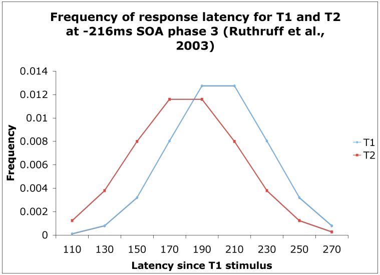
Ruthruff et al., 2003 report a PRP experiment with:

- Single participant.
- Unordered responses.

Now imagine if subject must produce **ordered** responses:

- At long SOAs no SRD is required to avoid response reversal.
- At short SOAs more than 50% response reversal when objective not sensitive to reversal.



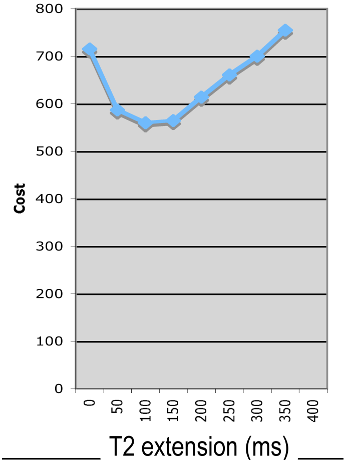


A Very Simple Constraint Model

- Constraints consist of the mean overall RTs and SDs.
- Space of strategies defined by a single variable: Extension of T2 response (E).
  - A simple form of Meyer and Kieras' SRD
- Objective is to minimize duration and response reversals.
  - Note the **trade-off:** Reduced reversals vs. total duration.

Combining Task + Architecture to Compute Optimal Behavior

- Now we can compute cost function from Monte-carlo simulations given this subject's standard deviation of RTs.
- Note that this combines two features:
  1. Constraints on the **TASK** (ordering and speed constraints, as expressed through explicit payoff).
  2. Constraints on the **ARCHITECTURE** (noise in the performance system).



Simple Ordered Responses

New experiments (Kopecky):

1. Visual cue appears.
2. Subject must quickly press two keys in order:
   - Left index, right middle.
   - Left middle, left index.
   - Right ring, left middle.
   - etc.
3. Subject rewarded for speed and accuracy.

SIMPLE MODEL

- Subjects defer R2 for *IRI* milliseconds after R1, where IRI maximizes payoff given their indiosyncractic variance.

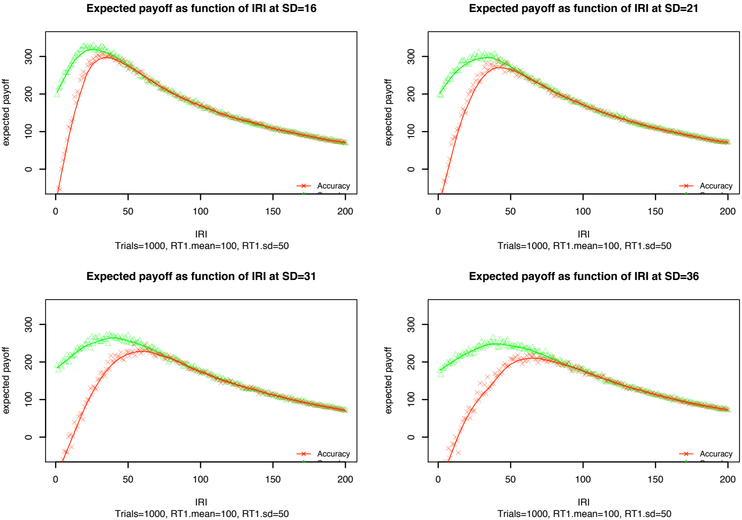
Explicit Payoff Schemes

Subjects were rewarded with CASH with explicit payoff schemes. Example:

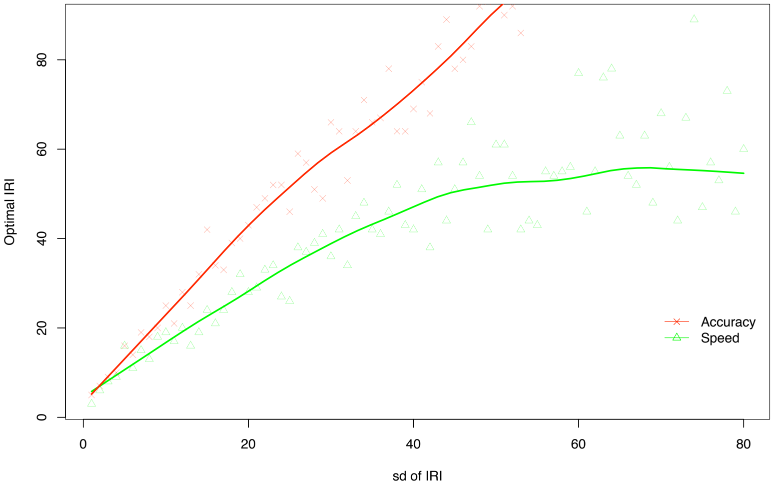
- If correct and Total RT $< 500ms$, then award $100 - RT/5$ points.
- If correct and Total RT $>= 500ms$, then award zero points.
- If incorrect, then lose 100 points.

Sample Payoff Curves at 4 Different Standard Deviations (SD) of IRI

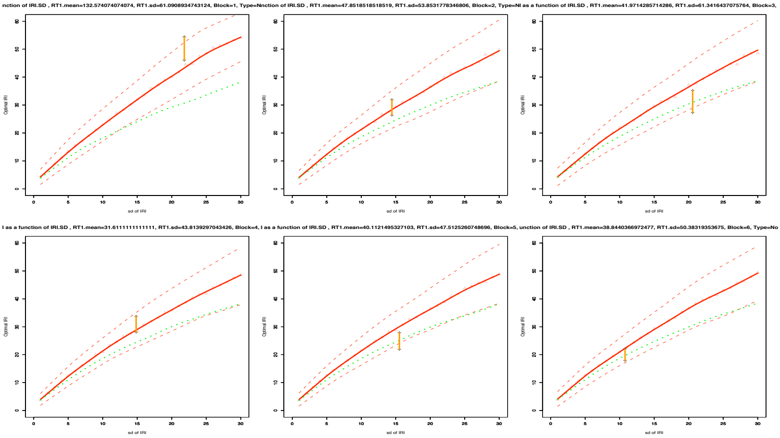
Sample Payoff Curves at 4 Different Standard Deviations (SD) of IRI



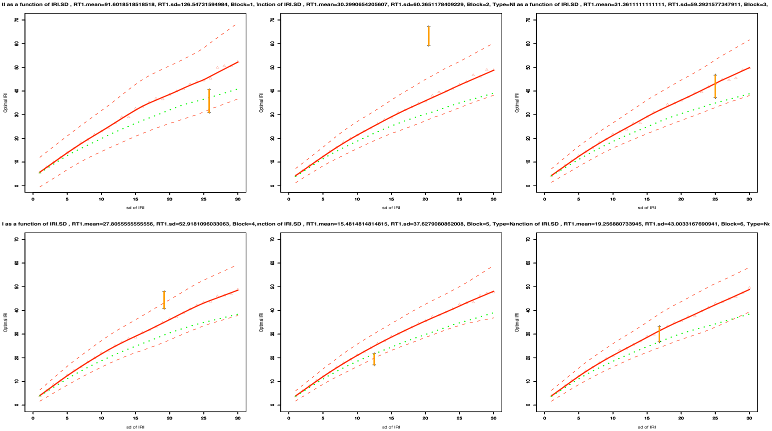
Predicted optimal IRI as function of SD of IRI



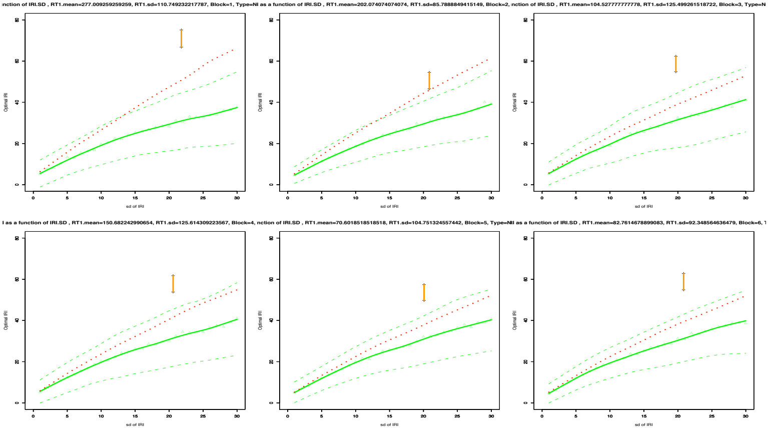
A Good Subject



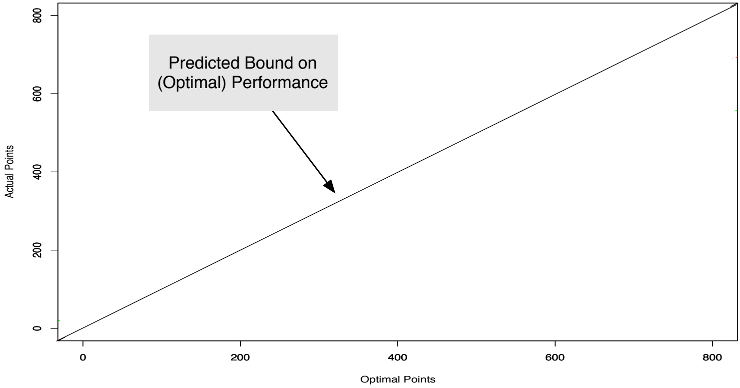
Another Good Subject



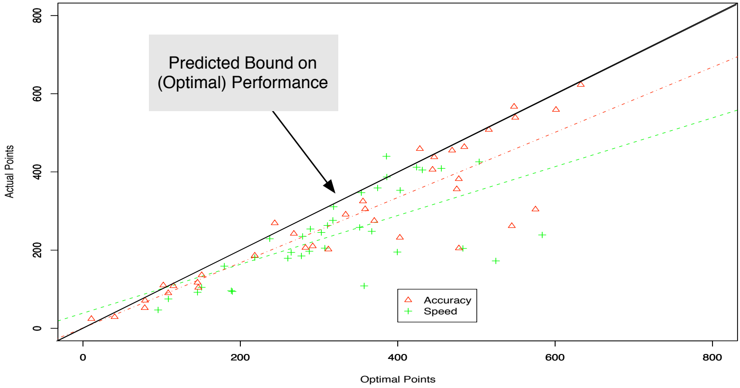
A So-So Subject



All Subjects, all Finger-Pairs: Actual vs. Predicted Optimal Points



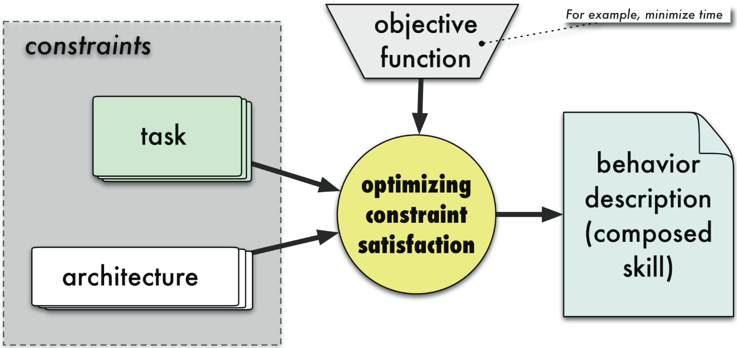
All Subjects, all Finger-Pairs: Actual vs. Predicted Optimal Points



This was a simple task where the strategy space = single quantitative variable.

Analysis of more sophisticated strategies needs a more general solution. . .

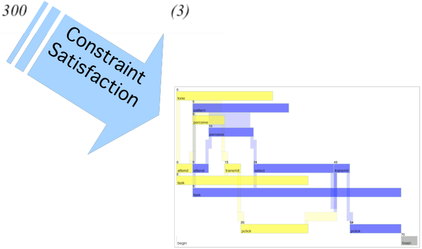
Cognitive Constraint Modeling



Cognitive Constraint Modeling

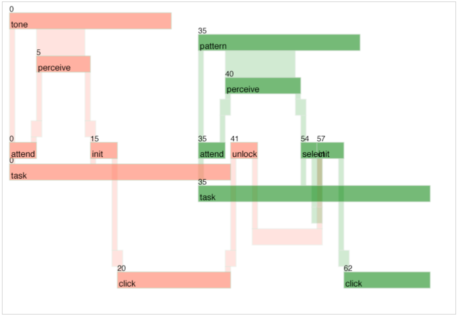
The tool is called CORE: Constraint-based Optimizing Reasoning Engine

$$\begin{aligned}
&\forall P_j\text{: } \{\ (isa,process)\ (name,initclick) \\
&\qquad (start,S_j)\ (duration,D_j)\ \} \subseteq P_j \\
&\rightarrow \\
&\exists P_i\text{: } \{\ (isa,process)\ (name,click)\ (start,S_i)\ \} \subseteq P_i \\
&\wedge\ S_j + D_j \leq S_i \\
&\wedge\ S_i - (S_j + D_j) \leq 300 \qquad (3)
\end{aligned}$$



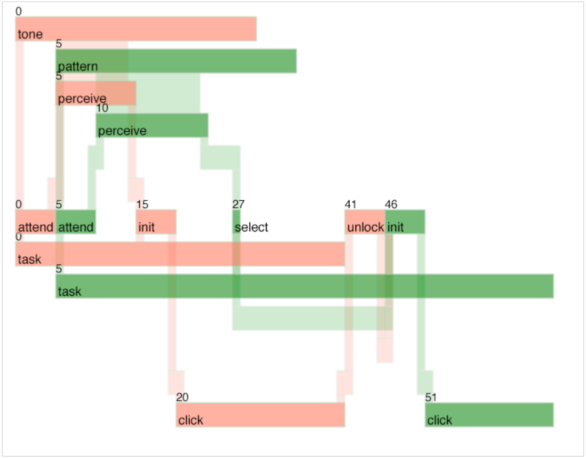
Constraints are logical relations between variables. They may specify partial values (e.g., duration, $D_i > 24$ ms), are non-directional (E.g. $S_j <= E_i + 300$ ms), and declarative.

Behavior Graphs



- Boxes represent **cascaded processes**.
- Rows of processes represent **resources** (cognition, perceptual, motor) and **world events**.
- Time is represented from left to right.
- Horizontal position represents onset.
- Spatial extent represents duration.

Dual-task (PRP) with simple set of process & information-flow constraints (50ms SOA)

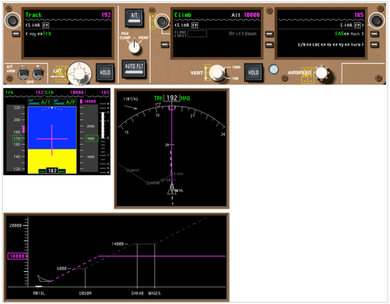


But does the approach scale to more complex tasks?

Comparing two cockpit designs

777

FDF



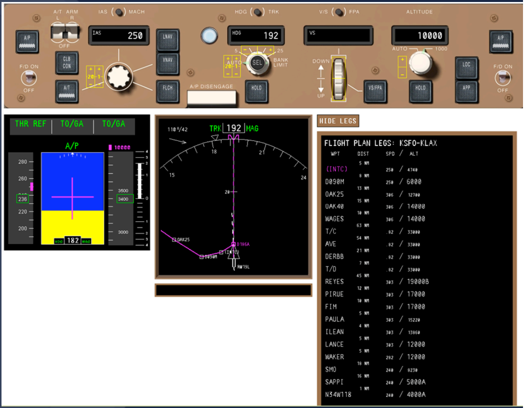
Goals:

- New design should reduce errors
- New design should be no slower than old

777 Interface: Task 1

"You are following the altitude restrictions of the Moorpark 3 arrival; your last altitude clearance was 1-7 thousand. Descend via the Moorpark 3 arrival; maintain 1-2 thousand"

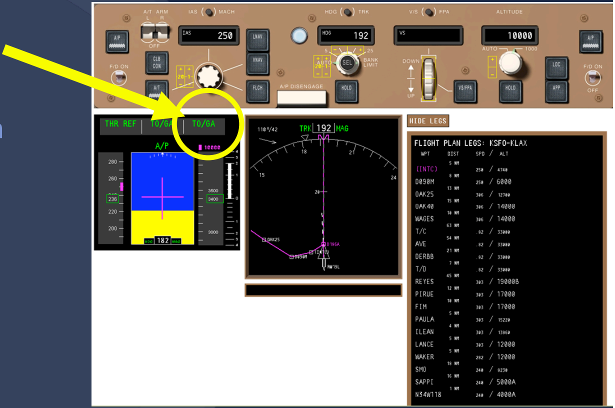
1. Verify current vertical mode
2. Dial Altitude Selector down to 12,000
3. Hit Altitude Selector
4. Verify new altitude
5. Verify new vertical mode



777 Interface: Task 1

"You are following the altitude restrictions of the Moorpark 3 arrival; your last altitude clearance was 1-7 thousand. Descend via the Moorpark 3 arrival; maintain 1-2 thousand"

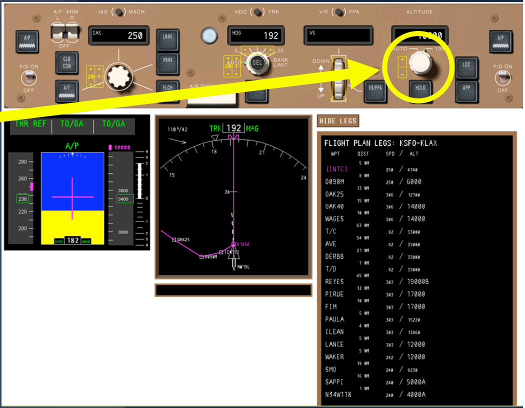
1. Verify current vertical mode
2. Dial Altitude Selector down to 12,000
3. Hit Altitude Selector
4. Verify new altitude
5. Verify new vertical mode



777 Interface: Task 1

"You are following the altitude restrictions of the Moorpark 3 arrival; your last altitude clearance was 1-7 thousand. Descend via the Moorpark 3 arrival; maintain 1-2 thousand"

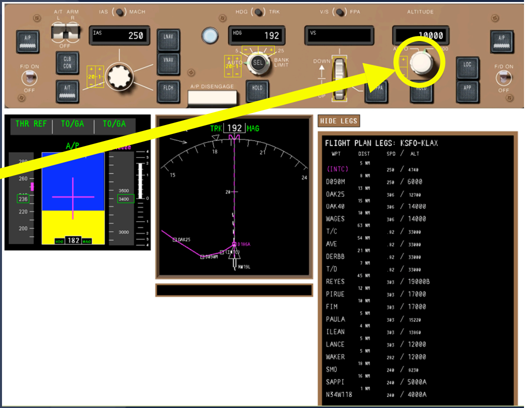
1. Verify current vertical mode
2. Dial Altitude Selector down to 12,000
3. Hit Altitude Selector
4. Verify new altitude
5. Verify new vertical mode



777 Interface: Task 1

“You are following the altitude restrictions of the Moorpark 3 arrival; your last altitude clearance was 1-7 thousand. Descend via the Moorpark 3 arrival; maintain 1-2 thousand”

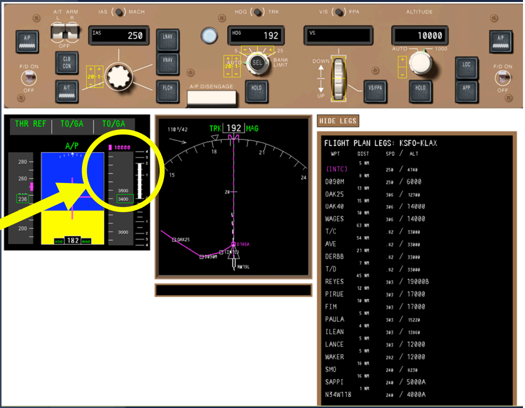
1. Verify current vertical mode
2. Dial Altitude Selector down to 12,000
3. Hit Altitude Selector
4. Verify new altitude
5. Verify new vertical mode



777 Interface: Task 1

"You are following the altitude restrictions of the Moorpark 3 arrival; your last altitude clearance was 1-7 thousand. Descend via the Moorpark 3 arrival; maintain 1-2 thousand"

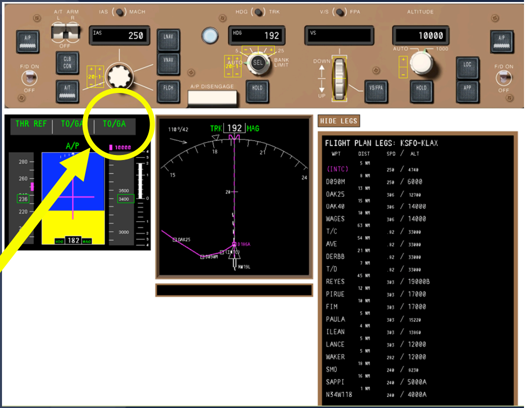
1. Verify current vertical mode
2. Dial Altitude Selector down to 12,000
3. Hit Altitude Selector
4. Verify new altitude
5. Verify new vertical mode



777 Interface: Task 1

"You are following the altitude restrictions of the Moorpark 3 arrival; your last altitude clearance was 1-7 thousand. Descend via the Moorpark 3 arrival; maintain 1-2 thousand"

1. Verify current vertical mode
2. Dial Altitude Selector down to 12,000
3. Hit Altitude Selector
4. Verify new altitude
5. Verify new vertical mode



The Demands of Applied Modeling

- Not just interested in *time*, but **memory load** and **ability to handle interruption**
- Tracking memory load requires specifying what must be held in memory and when
- Our task specification language and models capture this in the form of **information flow constraints**. . .

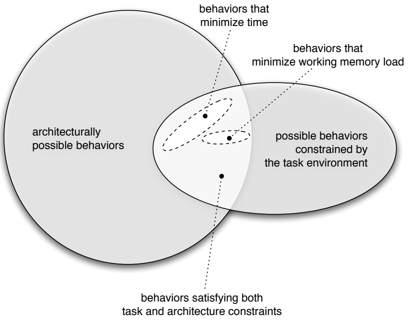
A Natural Task Specification

Captures the task's **information flow** rather than a fixed sequence of steps.

boeing FDF tt1
—>
comprehend situation : *FLIGHT_PLAN LAST_CLEARANCE*,
comprehend clearance : *INSTRUCTION* ***ALTITUDE***,
get vertical_mode after INSTRUCTION ALTITUDE : *VMODE*,
set altitude to **ALTITUDE** given INSTRUCTION VMODE : *DIALED PUSHED*,
check limit against ALTITUDE after DIALED PUSHED : *LMT_CHECKED*,
check ap_status against INSTRUCTION after LMT_CHECKED : *AP_CHECKED*.

Emergent Strategies

- Fully specified task constraints may still leave many details of behavior unspecified
- These details are automatically worked out by CORE to satisfy the architectural constraints
- *Example:* Precise timing of the perception of the mode information



Emergent Strategies

Early look to the mode display, in series with the rest of the task:

Emergent Strategies

Outline

Overview of Constraint Analysis

Example #1: Dual-tasks

How It Works

Example #2: 777 Cockpit

Example #3: ACT-R Critique

Summary

Later look to the mode display, in parallel with dialing the altitude:

A PRP Emergency!!

WHAT IF . . . the pilot had to handle an auditory interruption that required a manual button press response?

boeing FDF tt1

→

comprehend situation
comprehend clearance
get vertical_mode
set altitude
check limit
check ap_status

auditory interruption

→

auditory tone,
attend auditory
perceive auditory tone,
choose_response
press key.

A PRP Emergency!!

WHAT IF . . . the pilot had to handle an auditory interruption that required a manual button press response?

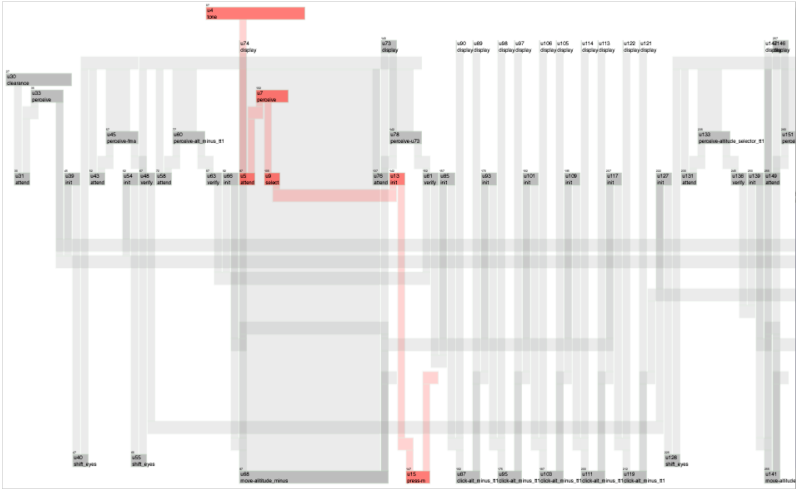
task

$\longrightarrow$

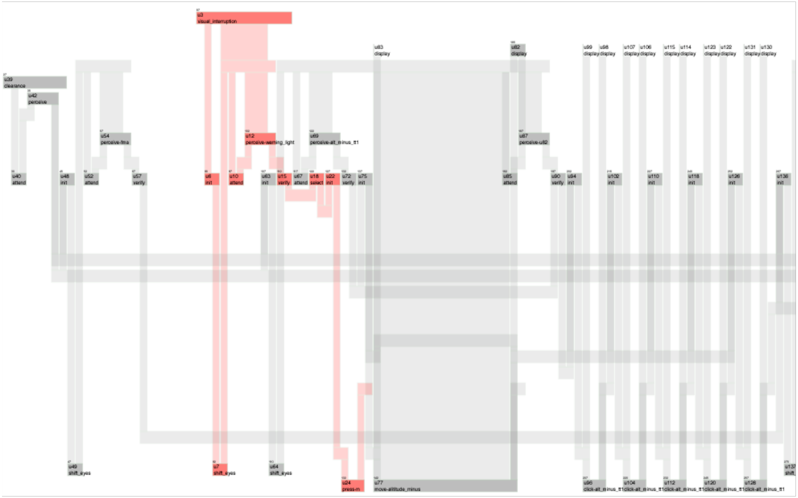
auditory interruption,

boeing FDF tt1.

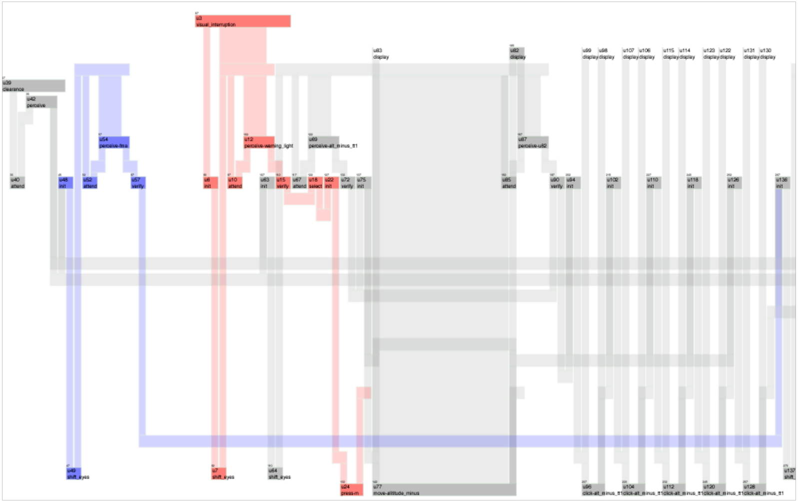
Auditory Interruption



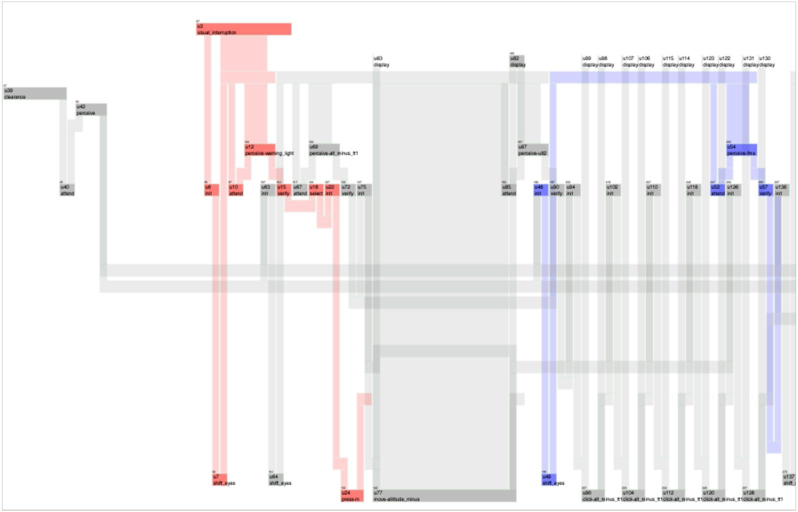
Visual Interruption



Visual Interruption



Same Task Spec, Different Objective: Reduce Memory Load



24 Models

Alina Chu (UM) and Katherine Eng (NASA)

2 interfaces {FDF, 777} × 2 tasks × 3 interruption conitions × 2 optimizations {time, WM}

Interesting predictions:

1. FDF faster than 777
2. Little difference in WM load
3. Simple auditory interruption slightly increases time and WM load
4. Simple visual interruption increases time more, and effect is greater for 777

But is this just a different way to do architectural modeling, or does it really change the way we should build and test cognitive models?

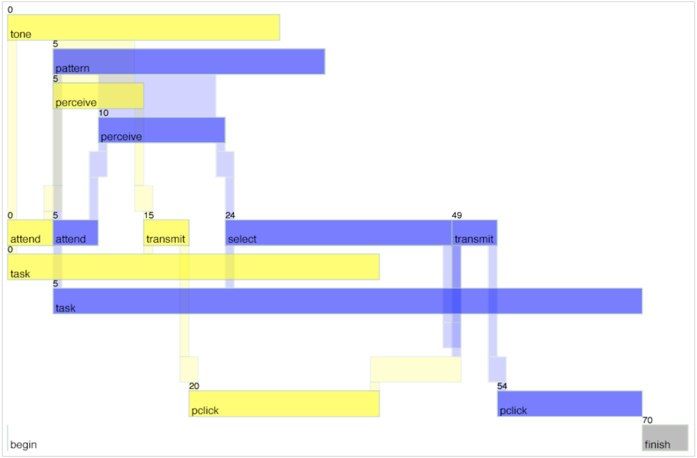
ACT-R vs. EPIC, in Psych Review (2001)

- In a prominent article promoting **an ACT-R account of PRP effects**, Byrne & Anderson (2001) created models that exhibited a *dual-task interference effect* based on ACT-R's theory of memory activation.
  - In ACT-R, retrieval time from memory is sensitive to a limit on total source activation.
  - The more retrieval features on the goal, the less activation each features receives.
- The models exhibited a dual-task interference effect because the source activation was less when tasks overlapped and the goal contained features from both tasks.

We can model this as a constraint.

An ACT-R model of the PRP task

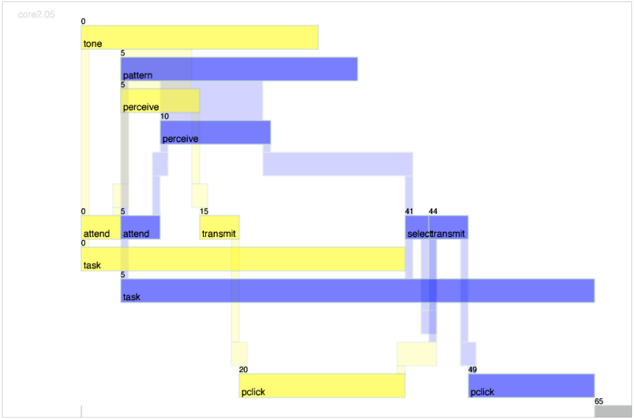
Our reconstruction of one of the models in Byrne & Anderson (2001), *Psychological Review*:



What did the original ACT-R model explain?

- Byrne and Anderson created several ACT-R models (based on particular strategies) that fit the data.
- **But if a better strategy is available, given ACT-R's constraints, has skilled PRP performance been explained?**

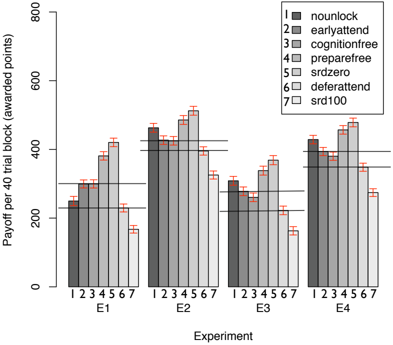
ACT-R model (optimal)



The optimal model not only deferred response, **but deferred retrieval too.** Byrne and Anderson didn't think of this—and neither did we.

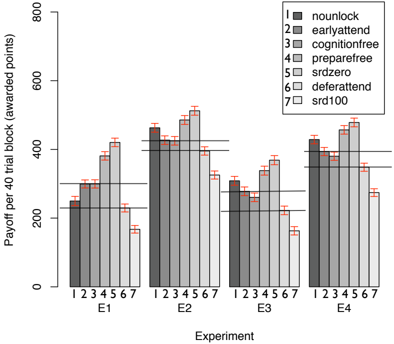
An Astonishing Result

Using CORE, we performed a systematic analysis of possible strategies for ACT-R models on all four PRP experiments modeled in Bryne & Anderson (2001), computing the expected payoff based on 40,000 runs.



An Astonishing Result

Using CORE, we performed a systematic analysis of possible strategies for ACT-R models on all four PRP experiments modeled in Bryne & Anderson (2001), computing the expected payoff based on 40,000 runs.



In each experiment, the Byrne & Anderson models consistently underperform—sometimes by substantial amounts—the best strategy.

Summary/Nuggets de Oro

1. Adaptation is bounded by the task environment and architecture.
2. An architecural theory explains behavior, with no further assumptions, if the optimal performance predicted by the theory corresponds to the observed asymptotic bound.
3. **Constraint satisfaction can be used to predict the asymptotic bound on adaptation, formally deriving the predictions of an architectural theory while minimizing assumptions about strategy.**
4. **Significant theoretical and applied benefits may accrue from this approach and its associated tools.**

Nuggets de Carbón

1. **Efficiency.** Some models take 2 seconds, some take 24 hours, some never return.
2. **Interaction with task simulation.** Presently, can't be done.
3. **Difficulty formalizing learning constraints.** Presently, can't be done (though we haven't really tried).

Acknowledgements

People:

- Alina Chu (Michigan)
- Katherine Eng (NASA Ames)
- Jonathon Kopecky (Michigan)
- Juliet Richardson (Convergys, UK)
- Mason Smith (Michigan)
- Irene Tolinger (NASA Ames)

Agencies/companies:

- Office of Naval Research
- NASA (Ames Research Center)
- Boeing