The Shape of Things to Come: An Emerging Constellation of Interconnected Tools for Developing the Right Cognitive Model at the Right Scale

Wayne D. Gray
Professor of Cognitive Science
Professor of Computer Science

Outline

• Discussion of problems with existing technologies for computational cognitive modeling
• Introduction of three very different tools
• Discussion of their interconnectedness
• The shape of things to come
Major Problems with the Current Tools for Cognitive Modeling

- Ease of Use
- Toolbox
- Time Scale
- Generative Models
- Emergence
Major Problems: Ease of Use

- Prerequisite Knowledge
  - Computer Science
  - Cognitive Science
  - Domain Expertise

- Building Models
  - 80/20 rule
    - In large models of complex task, writing 80% of the code feels as tedious as writing in assembly language
    - But, cannot get to the interesting 20% unless the other 80% is written

- Assessment: Modeling is too hard and takes too long
Major Problems: Toolbox

- An ideal toolbox would have a perfect tool for each task
- Too many modelers, too much of the time, act as if the only tool in their toolbox was a hammer
- Assessment: *To a man with a hammer, everything looks like a nail* - there are a lot of hammers out there!
## Major Problem: Time Scale

Newell’s Analysis of Levels: The Time Scale of Human Action

<table>
<thead>
<tr>
<th>Scale (sec)</th>
<th>Time Units</th>
<th>System</th>
<th>Analysis</th>
<th>World (theory)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10000000000</td>
<td>decades</td>
<td>Technology</td>
<td>Culture</td>
<td>Social &amp; Organizational</td>
</tr>
<tr>
<td>100000000</td>
<td>years</td>
<td>System</td>
<td>Development</td>
<td></td>
</tr>
<tr>
<td>10000000</td>
<td>months</td>
<td>Design</td>
<td>Education</td>
<td></td>
</tr>
<tr>
<td>1000000</td>
<td>week</td>
<td>Task</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10000</td>
<td>days</td>
<td>Task</td>
<td>Traditional Task Analysis</td>
<td>Bounded Rationality</td>
</tr>
<tr>
<td>1000</td>
<td>hours</td>
<td>Task</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>10 min</td>
<td>Task</td>
<td></td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>min</td>
<td>Subtask</td>
<td>Strategies &amp; Procedures</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>10 sec</td>
<td>Unit task</td>
<td>Procedures &amp; Methods</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1 sec</td>
<td>Interactive Routines</td>
<td>Embodiment Level (½ to 3 s)</td>
<td>Cognitive Band (symbolic)</td>
</tr>
<tr>
<td>0.1</td>
<td>100 ms</td>
<td>Production System</td>
<td>Elements (DME-MA-VA)</td>
<td></td>
</tr>
<tr>
<td>0.01</td>
<td>10 ms</td>
<td>Atomic</td>
<td>Architectural</td>
<td>Biological Band (subsymbolic)</td>
</tr>
<tr>
<td>0.001</td>
<td>1 ms</td>
<td>Parameters</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>


Major Problems: Time Scale

- Human behavior transcends multiple time scales of human activity
- To understand complex behavior it may be productive to model at multiple time scales
- Example: There are many models of eye movements made during reading. To a nonmodeler interested in understanding reading comprehension, it might seem reasonable to scale up such a model to predict reading comprehension problems among children
- Assessment: Little reuse of models built to capture one time scale to modeling phenomena that are emergent at other time scales
  - Time scale intolerance and a concomitant lack of time scale interconnectedness
Major Problems: Generative & Predictive

- Static and Descriptive
  - Too many models simply trace the cognitive, perceptual, and motor activities required to account for one particular instance of one task

- Generative and Predictive
  - We want models that generate predictions in dynamic task environments where the exact order of events is not determined and the detailed features of the task environment may not be fixed

- Assessment: Limited generativity. Small changes to the task or task environment often require editing or major rewrites of model
Major Problems: Emergence

- We would like to start our models with the basic facts or strategies of the novice performer and predict changes over time with practice and experience
  - Speed up performance
  - Optimization of strategy selection
  - Emergence of new strategies
  - Emergence of new declarative knowledge
Major Problems: Emergence

- **Assessment:**
  - **Cognitive modeling approaches** (e.g., architectures, reinforcement learning, Bayesian modeling) do an adequate job of speed up and optimization of a fixed set of knowledge and strategies
  - *Almost no discovery of new strategies with practice*
  - *Limited emergence of declarative knowledge - new facts, but not new types/categories*
Major Problems with the Current Tools for Cognitive Modeling

- Why are these issues important?
- An increasing recognition that human modeling of all types is important to our society and to national defense by enabling
  - The design of human-machine and human-information systems tailored to the capabilities of the human
  - Predictions of the reactions of populations to events
  - Network Science
  - Better training
  - Tailoring system to different populations of users (e.g., aging)
Outline

- Discussion of problems with existing technologies for computational cognitive modeling
- Introduction of three very different tools
- Discussion of their interconnectedness
- The shape of things to come
Three Tools: LARGE CAVEAT

• The title of the talk is *The Shape of Things to Come*

• No assertion that

  • The tools I mention are *THE TOOLS OF THE FUTURE*

• Rather, the intended claim is much more modest; namely,

  • The level of ease, flexibility, and interconnectedness that these tools aspire to are what we must work towards building
Descriptions and Report Cards for Three Tools

- ACT-R
- CogTool
- SANLab
A community of approximately 200 modelers at 91 institutions worldwide
Intentional Module (not identified)

Goal Buffer (DLPFC)

Visual Buffer (Parietal)

Visual Module (Occipital/etc)

Declarative Module (Temporal/Hippocampus)

Retrieval Buffer (VLPFC)

Manual Buffer (Motor)

Manual Module (Motor/Cerebellum)

External World

Anderson et al. (2004)
Waiting for ACT-R to connect
报告卡：ACT-R

- 便捷性
  - 必备知识：D
    - 最好拥有计算科学的副业
    - 认知科学的硕士学位或更好
    - & 领域专长

构建模型：D+

- 大约和使用任何编程语言构建任何大型软件系统一样容易。近年来，专门的工具被开发出来以辅助编写、调试和编辑模型。
Report Card: ACT-R

- **Toolbox:** C
  - **Limited Toolbox**
    - Many “closed form” ACT-R models written in spreadsheets to focus on single issues (e.g., memory, learning) that can be resolved by parameter fitting
    - Full models that actually perform the task using the same software as people use
  - **SAL - Synthesis of ACT-R & Leabra**
    - Possible but required resources of two extremely strong laboratories (Anderson/Lebiere at CMU and O’Reilly’s at CU)
Report Card: ACT-R

• Time Scales: C
  • Carnegie Tutors are written in a pre-ACT-R language that (among other things) incorporate much larger temporal steps than ACT-R (3 s versus 50 ms)
  • Is possible to write different models to focus on different levels of analysis, but this requires much advanced planning and is typically not done
Report Card: ACT-R

- Generative: $B^+$
  - ACT-R models can adapt to a range of changes in the task environment if the strategies built into the models can be applied

- Emergence:
  - Speed up with practice: $A$
  - Optimization of strategy selection: $A$
  - Emergence of new strategies: $F$
  - Acquisition of new declarative knowledge: $C$
    - (New facts, not new types)
Descriptions and Score Cards for Three Tools

- ACT-R
- CogTool
- SANLab
CogTool: Build a Model without Knowing Anything about Modeling

- Developed by Bonnie John (CMU)
  - http://cogtool.hcii.cs.cmu.edu/
- Target audience is HCI designers
- Storyboarding as modeling
Create Storyboards

- Capture screenshots from existing applications or create your own storyboards (e.g., PhotoShop™, etc)
- Example: Modeling the time that an experienced user would require to add a recurrent event to Google Calendar using one of the several methods provided by Google
Add hot spots
Collect a Bunch of Screen Shots, Add Hot Spots, and Link
Run Model
Run Model
Run Model
Run Model
Run Model

![Image of a computer window with a script for running a model]
Run Model
Run Model
Run Model

- Predicted expert performance time using this method to add a recurrent calendar event is: 20.342 s
Report Card: CogTool

• Ease of Use
  • Prerequisite Knowledge: A+
    • Ability to read a manual and experience with modern graphic interfaces
    • Domain Expertise
  • Building Models: A+
    • Very easy: More like storyboarding than modeling
Report Card: CogTool

• Hammer versus Toolbox
  • A hammer not a toolbox: F
  • Keystroke Level GOMS - Predictions of Expert Performance Time
Report Card: CogTool

- Time Scale: F
  - Ability to go up and down a level of analysis easily
  - One Level
Report Card: CogTool

- Generative: D
  - Minor edits are easy, but no true generative capability
  - If environment changes model must be edited to reflect those changes

- Emergence: F
  - No speed up in performance, no optimization of strategy selection, no new strategies, no new declarative knowledge
Descriptions and Score Cards for Three Tools

- ACT-R
- CogTool
- SANLab
SANLab

- The Stochastic Analytic Network Laboratory (SANLab) for Cognitive Modeling
- Developed by Wayne Gray & Evan Patton (RPI)
Activity Networks

- Activity networks are directed, acyclic graphs where each node in the graph represents some process in time, and connections between nodes represent execution order.
- Used in PERT charts for project management.
Activity Networks

- Activity Network Modeling in Cognitive Science
  - First applied to cognitive modeling by Schweickert (1978; 1980)
  - John & Gray’s work with CPM-GOMS (Project Ernestine, Milliseconds Matter)

Activity Networks

• Activity Networks provide a network of dependencies with the longest path through that network being named the Critical Path
  
  • That is, the shortest time an activity can complete is determined by the longest path through the network
What a Critical Path Analysis can Tell You: Beginning

What a Critical Path Analysis can Tell You: Ending

Most cognitive activity network models use mean times for operations (i.e., constant time)

Schweickert and colleagues point out that cognitive operations have variability and that this variability can result in multiple critical paths


Constructing a very simple CPM-GOMS model in SANLab

- Parts
- Interleaving
- Stochasticity
- Comparison of very simple models
Building a Preliminary CPM-GOMS Model

CPM-GOMS Templates

Perceive Simple Sound

Perceive Visual Information With Eye Movement
Building a Preliminary CPM-GOMS Model
Cut & Paste & String Together
Building a Preliminary CPM-GOMS Model
Run Model Once to Generate Critical Path

Red outlined boxes show the critical path: note that ALL operations are on the critical path!

PO (A): perceive sound (x)
Constant 100

PO: Perceive Visual
Constant 0

AO: attend sound (x)
Constant 50

VO: verify sound (x)
Constant 50

CO: Initiate Eye Movement EyeMvt
Constant 50

CO: Verify EyeMvmt
Constant 50

MO: Eye Movement EyeMvmt
Constant 30

Estimate time = 670 ms
Two Problems:

- First: Assumes seriality – all operations of subtask A are completed before subtask B can begin
- FALSE: at this level of analysis the human cognitive controller can interleave subtasks
Building a Preliminary CPM-GOMS Model

Interleave Operators

Estimate time
= 620 ms

Critical path no longer goes through ALL operators

Interleaved operators
Two Problems

- First: Assumes seriality – all operations of subtask A are completed before subtask B can begin
  - FALSE: at this level of analysis the human cognitive controller can interleave subtasks

- Second: Assumes constancy of performance time for individual operations as well as for the entire task
  - FALSE: process times for human cognitive, perceptual, and motor operations are stochastic, not constant
Building a Preliminary CPM-GOMS Model
Interleave Operators + Stochastic Operation Times

Gamma Distribution
(randomly sampled on each model run)
Interleave Operators + Stochastic Operation Times, then Run the Model 10,000 times

Two critical paths!
Different mean times!
Two Critical Paths

Mean time: 635 ms

Mean time: 598 ms
### Very Simple Model: Summary

<table>
<thead>
<tr>
<th>Interleaving</th>
<th>Constant/Stochastic</th>
<th>Critical Path</th>
<th>Predicted Times</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Interleaving</td>
<td>Constant</td>
<td>One</td>
<td>670 ms</td>
</tr>
<tr>
<td>Interleaving</td>
<td>Constant</td>
<td>One</td>
<td>620 ms</td>
</tr>
<tr>
<td>Interleaving</td>
<td>Stochastic</td>
<td>Average</td>
<td>628 ms</td>
</tr>
<tr>
<td>Interleaving</td>
<td>Stochastic</td>
<td>79%</td>
<td>635 ms</td>
</tr>
<tr>
<td>Interleaving</td>
<td>Stochastic</td>
<td>21%</td>
<td>598 ms</td>
</tr>
</tbody>
</table>
What Does Stochasticity Add?

- Created SANLab models for each of the original Project Ernestine models
- Ran the SANLab models 10,000 times each
  - Compared the fit of the original constant time model versus same models with stochastic times
  - Looked at the variability across critical paths
Fit of Original Constant Models versus Stochastic Models

Old Workstation

Constant

Stochastic

$R^2 = 0.716$

$R^2 = 0.741$
Predicted Variability Across Critical Paths for 6 Call Types

Critical Paths for Six Models

cc01

cc02

cc03

cc04

cc05

cc06
Implications of Adding Variability to Model Runs

- How long does *normal* performance take?
  - When stochasticity is added we gain predictions of the variability in normal performance
  - Are we planning for normal variability? Or are we expecting a constancy in real-time safety-critical human performance that does not exist?
- The variability produced by SANLab is in the absence of stress, fatigue, or danger - variability would only increase under those circumstances
Report Card: SANLab

- Ease of Use
  - Prerequisite Knowledge: B
    - No computer science, but some sophistication in cognitive science (Masters? PhD?)
  - Building Models: B-
    - Easy (drag and click interface)
    - Large models can be tedious to build
    - Tools are being added to facilitate comparisons of critical paths and other measures
Report Card: SANLab

- Hammer versus Toolbox
  - A hammer not a toolbox: F
  - CPM-GOMS level of analysis
Report Card: SANLab

• Time Scale: $D^+$

• One Level

• But, can inspect the model at various levels (e.g., total performance time, operations on the critical path, number of critical paths and differences among them, etc)
Report Card: SANLab

- Generative: D
  - Minor edits are easy, but no true generative capability
  - If environment changes model must be edited to reflect those changes

- Emergence: D
  - No speed up in performance, no optimization of strategy selection, no new strategies, no new declarative knowledge
  - Can discover new Critical Paths
    - Key feature of SANLab is the distinction it makes between variations on a single strategy (multiple critical paths) versus variations between different strategies
# Report Card

<table>
<thead>
<tr>
<th></th>
<th>Ease of Learning and Use</th>
<th>Toolbox versus Hammer</th>
<th>Time Scales</th>
<th>Generative</th>
<th>Emergence</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Prereq Knwldg</td>
<td>Bldg Models</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>ACT-R</strong></td>
<td>D</td>
<td>D⁺</td>
<td>C</td>
<td>C</td>
<td>B⁺</td>
</tr>
<tr>
<td><strong>CogTool</strong></td>
<td>A⁺</td>
<td>A⁺</td>
<td>F</td>
<td>F</td>
<td>D</td>
</tr>
<tr>
<td><strong>SANLab</strong></td>
<td>B</td>
<td>B⁻</td>
<td>F</td>
<td>D⁺</td>
<td>D</td>
</tr>
</tbody>
</table>

Tuesday, April 6, 2010
Outline

• Discussion of problems with existing technologies for computational cognitive modeling
• Introduction of three very different tools
• Discussion of their interconnectedness
• The shape of things to come
CogTool to ACT-R

- CogTool to ACT-R
  - CogTool derives its KLM predictions from a simplified version of ACT-R
  - Running a CogTool model creates a simple ACT-R model that is run to generate an ACT-R trace. The KLM times are based on this trace
- Good use of harnessing the power of ACT-R to generate a simplified model
  - It may be possible to use CogTool to generate much of the repetitive detail in an ACT-R model & then allow the modeler to concentrate on the rest of the model
ACT-R to SANLab

- It is possible to import the trace of an ACT-R model into SANLab (Patton, 2009)
- This would allow you to vary the parameters (mean times and distribution) of one run of the model to determine how different settings would affect the outcome
- This would also alleviate much of the tedium of building SANLab models
CogTool to ACT-R to SANLab

- It is possible (Patton, 2010) to export the ACT-R trace of a CogTool model into SANLab
Status of Tools

- Publicly Available:
  - ACT-R: [http://act-r.psy.cmu.edu/](http://act-r.psy.cmu.edu/)
  - Cogtool: [http://cogtool.hcii.cs.cmu.edu/](http://cogtool.hcii.cs.cmu.edu/)

- The ACT-R to SANLab connection is experimental and is not released
- The CogTool to SANLab (via ACT-R) connection is also experimental and not released
Outline

- Discussion of problems with existing technologies for computational cognitive modeling
- Introduction of three very different tools
- Discussion of their interconnectedness
- The shape of things to come
CogTool ↔ [ACT-R] ↔ SANLab

• Imagine two types of users
  • Those who harness the power of ACT-R to go from CogTool to SANLab, but who do not know ACT-R
  • Those who freely use the power of the entire system to facilitate their work on each of these three

• How might the combination of these three systems score on our criteria?
### Report Card: CogTool ↔ ACT-R ↔

#### Ease of Use

<table>
<thead>
<tr>
<th></th>
<th>ACT-R</th>
<th>CogTool</th>
<th>SANLab</th>
<th>CogTool ↔ SANLab</th>
<th>CogTool ↔ SANLab ↔ ACT-R ↔</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Prerequisite Knowledge</strong></td>
<td>D</td>
<td>A+</td>
<td>B</td>
<td>B</td>
<td>D</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>MS in CogSci</td>
<td>Minor in CompSci, MS in CogSci</td>
</tr>
<tr>
<td><strong>Building Models</strong></td>
<td>D+</td>
<td>A+</td>
<td>B-</td>
<td>A+</td>
<td>B</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Most SANLab models can be started in CogTool</td>
<td>If can export to ACT-R from both SANLab and CogTool</td>
</tr>
</tbody>
</table>
Hammer versus Toolbox

<table>
<thead>
<tr>
<th>ACT-R</th>
<th>CogTool</th>
<th>SANLab</th>
<th>CogTool ⇔ SANLab</th>
<th>CogTool ⇔ SANLab ⇔ ACT-R ⇔</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>F</td>
<td>F</td>
<td>C</td>
<td>B</td>
</tr>
</tbody>
</table>
## Report Card: CogTool ↔ ACT-R ↔

### Time Scales

<table>
<thead>
<tr>
<th>ACT-R</th>
<th>CogTool</th>
<th>SANLab</th>
<th>CogTool ↔ SANLab</th>
<th>CogTool ↔ SANLab ↔ ACT-R ↔</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>F</td>
<td>D+</td>
<td>C</td>
<td>B</td>
</tr>
</tbody>
</table>

Tuesday, April 6, 2010
Report Card: CogTool ↔ ACT-R ↔

- Generative

<table>
<thead>
<tr>
<th>ACT-R</th>
<th>CogTool</th>
<th>SANLab</th>
<th>CogTool ↔ SANLab</th>
<th>CogTool ↔ SANLab ↔ ACT-R ↔</th>
</tr>
</thead>
<tbody>
<tr>
<td>B+</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>B+</td>
</tr>
</tbody>
</table>
### Emergence

<table>
<thead>
<tr>
<th></th>
<th>ACT-R</th>
<th>CogTool</th>
<th>SANLab</th>
<th>CogTool ↔ SANLab</th>
<th>CogTool ↔ SANLab ↔ ACT-R ↔</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Speed up with practice</strong></td>
<td>A</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>A</td>
</tr>
<tr>
<td><strong>Optimized Strategy Selection</strong></td>
<td>A</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>A</td>
</tr>
<tr>
<td><strong>New Strategies</strong></td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td><strong>New Declarative Knowledge</strong></td>
<td>C</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>C</td>
</tr>
<tr>
<td><strong>New Critical Paths</strong></td>
<td>F</td>
<td>F</td>
<td>A</td>
<td>A</td>
<td>A</td>
</tr>
</tbody>
</table>
The Shape of Things to Come???

- Conclusions
  - A small toolbox
  - Far from complete
  - But . . . the combination of the three is more powerful than any one by itself
  - Maybe, this is the shape of things to come . . .
  - Clearly need more work on integrating these tools, adding other tools to this toolkit, and creating different toolkits that have different sets of tools
The Shape of Things to Come

- Problems in making the vision happen
  - CogTool has been guided by a vision of building cognitive science into the tools used by practitioners
  - SANLab has been guided by a desire to build an easy to use tool to do a certain type of cognitive modeling
  - ACT-R focus on theory not on applications
The Shape of Things to Come

- Lack of funding for tool building has hampered development
  - SANLab - rests almost entirely on the programming skills of one brilliant undergraduate who is now a brilliant graduate student
  - Little academic reward for building CogTool or SANLab
  - Some funding support for CogTool, none for SANLab
- The synergy discussed here is a pleasant by-product but was not anticipated by the creators
The Shape of Things to Come

- Together the three are greater than the sum of their parts
  - Ease of building models greatly increases if simple ACT-R models can be created from CogTool and SANLab models
  - Combining any two of these helps alleviate the hammer problem by creating a small toolbox
  - Alleviate the time scale issue
  - Generatively & Emergence are not synergistic over tools, but simply enable CogTool and SANLab models to benefit from ACT-R’s generally higher scores on these attributes