General Article

USING A COGNITIVE ARCHITECTURE TO EXAMINE WHAT DEVELOPS

By Gary Jones, Frank E. Ritter, and David J. Wood

ESRC Centre for Research in Development, Instruction and Training, School of Psychology, University of Nottingham, Nottingham, England

Different theories of development propose alternative mechanisms by which development occurs. Cognitive architectures can be used to examine the influence of each proposed mechanism of development while keeping all other mechanisms constant. An ACT-R computational model that matched adult behavior in solving a 21-block pyramid puzzle was created. The model was modified in three ways that corresponded to mechanisms of development proposed by developmental theories. The results showed that all the modifications (two of capacity and one of strategy choice) could approximate the behavior of 7-year-old children on the task. The strategy-choice modification provided the closest match on the two central measures of task behavior (time taken per layer, r = .99, and construction attempts per layer, r = .73). Modifying cognitive architectures is a fruitful way to compare and test potential developmental mechanisms, and can therefore help in specifying "what develops."

Examining how children's performance improves, and detailing what changes occur during development, is difficult, mainly because development occurs over a number of years. Observations spanning several years are necessarily sparse, which means that the mechanisms giving rise to the changes in performance have to be inferred (Siegler, 1995). Many theories of development are based on these sparse observations, so the developmental mechanisms they propose suffer problems of being ill-defined and hence difficult to test.

These problems can be addressed at a micro level by using the microgenetic method. This generally takes the form of intense study of task behavior over a period of weeks when cognitive change occurs within the task (Siegler & Jenkins, 1989). The microgenetic method offers valuable insights into cognitive change because of the extensive amount of data collected. The limitation of the method is that it covers only a small period of time, whereas development occurs over a number of years.

Computational modeling can help to both force precision and enable the testing of proposed theoretical mechanisms. Modeling enables the manipulation of single variables while reliably controlling for all others. A model can therefore be used to test the influence of one theoretical mechanism of development while keeping all other mechanisms constant. The programlike nature of computational models also forces the complete definition of any ill-defined concepts that are part of a theoretical mechanism of development.

Computational models have a long but not extensive history within development. Young (1976) and Klahr and Siegler (1978) modeled improvement in task behavior by increasing the knowledge base of their models. McClelland and Jenkins (1991) modeled development by repeatedly training their model on task problems, the repeated training corresponding to more task experience. Shultz, Schmidt, Buckingham, and Mareschal (1995) modeled development by giving the model more experience on the task, while allowing the model to alter its architecture.

Each of these models matched the data from children in its task. However, problems remain. First, development is believed to be more than simple increases in the knowledge base (Gentner, Rattermann, Markman, & Kotovsky, 1995). Second, development is not simply task experience (Elman et al., 1996). Third, the models match subject data on only a small number of measures, with every model omitting timing data. Timing data include both within-task times (the time taken to perform different components of a task or problem) and solution times (the time taken to perform the whole task or problem). No models have examined within-task times, although relative solution times were examined by Siegler and Shipley (1995) and by Siegler and Shrager (1984). Timing data are important because they can help highlight where learning takes place, and provide objective and precise comparisons between data from the model and from subjects.

The central mechanisms of development that have been proposed are knowledge changes[1] (e.g., Klahr & Wallace, 1976; Piaget, 1947/1950), increased capacity (Case, 1985; Halford, 1993; Pascual-Leone, 1969), strategy choice (Siegler & Shipley, 1995), and processing speed (Kail, 1991). Proponents of each mechanism generally admit that there are several

Address correspondence to Gary Jones, School of Psychology, University of Nottingham, Nottingham NG7 2RD, United Kingdom; e-mail: gaj@psychology.nottingham.ac.uk.

1. These include the changes in thinking and representation on which the Piagetian concept of stage is based.

Jones, G., Ritter, F. E., & Wood, D. J. (2000). Using a cognitive architecture to examine what develops. Psychological Science, 11(2), 93-100.

mechanisms at work; the mechanisms interact with each other to mediate development. For example, knowledge changes may be influenced by capacity (Case, 1985).

This article presents the first steps toward testing developmental mechanisms within a computational framework. Examining the influence of each mechanism of development within a single task will help clarify the roles each plays in cognitive development. The influence of each mechanism of development within a single task can be tested independently within a computational model of the task. We first outline a task to study development. Then we describe a computational model of the task, together with its fit to the behavior of adult subjects on the task. Finally, we present three example modifications to the model that independently manipulate two mechanisms of development: capacity and strategy choice (Jones, 1998, examined all four of the mechanisms listed in the previous paragraph). The discussion examines the results of the modifications and presents a program for future research.

A TASK TO STUDY DEVELOPMENT

The Tower task (Wood & Middleton, 1975) is to build a pyramid from 21 wooden blocks. Figure 1 shows the six layers to the pyramid; each of the lower five consists of four blocks, and a single block is the top layer. The blocks in the lower layers share the same characteristics, differing only in size. Each layer is normally formed via two sets of paired blocks. For example, as Figure 1 shows, placing the peg of Block A into the hole of Block B brings the two half-holes at their edges together to form a pair having a hole (a hole pair). Similarly, placing Blocks C and D together forms a pair with a peg (a peg pair). A layer is then formed by fitting the peg of the peg pair into the hole of the hole pair. The layers have interlocking circular indents above and circular depressions beneath, so that layers can be stacked to form the tower. The optimal number of constructions required to build the tower is 20, 3 for each of the lower five layers, and 5 to stack layers.

Typical errors include fitting half-pegs into holes, placing blocks flush to each other without having a peg-hole connection between them, and putting blocks of different sizes together. Incorrect constructions are rarely left uncorrected, because mismatches are usually readily apparent. It is natural to complete the tower layer by layer (all subjects in our experiments do this). Within-task learning can occur because the blocks in the lower five layers all share the same characteristics.

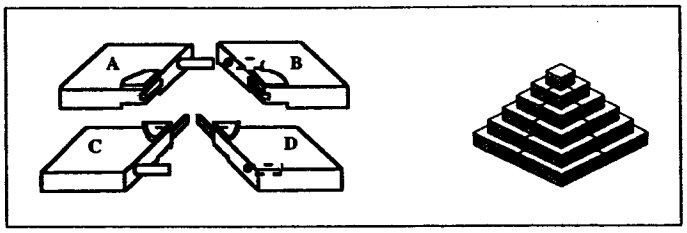

Fig. 1. Illustration of the Tower task. The blocks that make up a layer are shown on the left. Layers are stacked to create the tower shown on the right.

Why Use This Task?

The Tower task is ideal for studying development because it illustrates how children's problem-solving behavior changes across ages. Older children accomplish more correct constructions, produce fewer incorrect constructions, and take less time than their younger counterparts (Murphy & Wood, 1981, 1982; Wood, Bruner, & Ross, 1976; Wood & Middleton, 1975). The physical nature of the task enables detailed timing data to be recorded, and allows many strategies to be readily visible, reducing the need for the experimenter to infer mental structures and strategies.

Use of Cognitive Architectures

Cognitive architectures support the development of computational models within a common theoretical framework. The architectures usually incorporate general psychological mechanisms (e.g., the representation and characteristics of working memory) that constrain the way behavior can be modeled (e.g., Newell, 1990, chap. 1). These mechanisms include some of those proposed by developmental researchers.

Most current models of development have been successful in producing behavior at different levels of performance by adjusting only knowledge. Modifying knowledge alone does not reveal the extent to which other mechanisms may influence development. Cognitive architectures provide a principled way to examine the extent to which mechanisms other than knowledge may influence development.

In addition, most developmental models were not created within a cognitive architecture, so they could be manipulated in an unconstrained way in order to fit the subject data. Creating a model within a cognitive architecture forces constraints on the model. In particular, the timing predictions that architectures provide enable the temporal behavior of a model to be matched directly with that of subjects.

SIMULATING DEVELOPMENT

There are two natural ways to create a series of developmental models. One way is to model a lower performance level (that of children) and enhance the model to fit higher performance levels. The other way is to begin at the highest performance level (that of adults) and then degrade the model to fit lower performance levels. Adult performance on the Tower task is simpler to model because adult behavior is more regular, and adults can readily provide verbal protocols that help in the creation of a model. A model of adult behavior on the task

was therefore created first, and modifications to this model were compared with the behavior of children on the task.

The Task Simulation and Model of Adult Behavior

Both a task simulation and a model are necessary to study development in this task. The task simulation and model are detailed enough to allow modifications that represent the mechanisms proposed by developmental theories. A task simulation is required so that the task is appropriately represented and the time spent interacting with the blocks (i.e., looking at and manipulating the blocks) is properly estimated. Figure 2 shows the task simulation, which contains a full graphic representation of the task including all blocks and features.

The simulation includes an eye and two hands that are designed to meet a set of requirements identified for creating a psychologically plausible architecture for interacting with multiple external tasks (Baxter & Ritter, 1996). The architecture has been implemented several times in a variety of user interface management systems and graphics programming languages (Ritter, Baxter, Jones, & Young, in press).

The eye is able to saccade and fixate, and passes to the model what blocks and constructions it sees (e.g., a peg pair is represented as a construction of two blocks flush on their outer edges with their quarter circles and half-pegs aligned). The quality of the visual information passed to the model depends on where blocks are positioned in relation to the simulated eye. Three areas of decreasing visual quality are defined: fovea, parafovea, and periphery. The simulated hands are able to pick up, drop, rotate, turn over, fit, and disassemble blocks.

The cognitive aspects of the model are based on the ACT-R cognitive architecture (Anderson, 1993). ACT-R is a production-system-based architecture in which knowledge is represented as facts and rules. A rule has two sides, a condition and an action. When the conditions of a rule are met, its action can be taken (i.e., the rule can fire). Each condition normally corresponds to a fact that must be present within the system. Facts in ACT-R are called memory elements, with each memory element consisting of a number of attribute-value pairs.

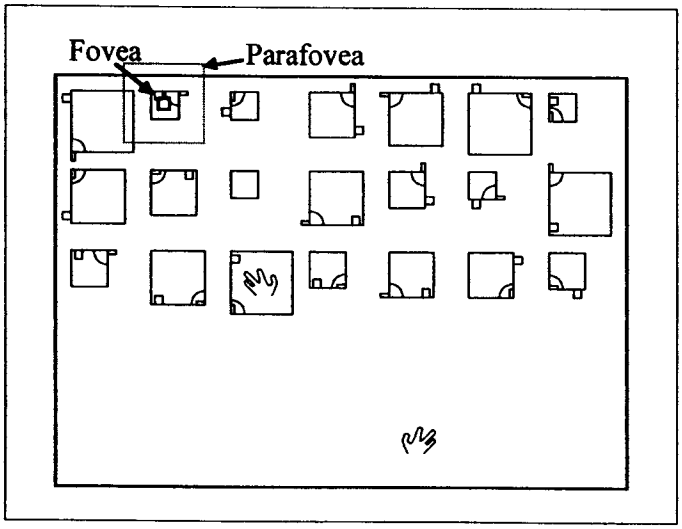

Fig. 2. Graphical simulation of the Tower task. The boundary rectangle represents a table top on which sit the 21 blocks (together with all block features) that make up the Tower task. The fovea and parafovea parts of the simulation eye are labeled, and the simulation hands are readily visible.

An example rule from the ACT-R model is shown in Table 1. The rule moves the left hand to a construction as a precursor to grabbing the construction. In order for the rule to fire, the system's current goal must be to grab a construction using the left hand. The left hand must not be near the construction, and the model's eye must be looking at the construction (so that the model knows where to move the left hand). If all these conditions are met, the hand is moved to the construction.

The adult model contains 317 ACT-R rules, including task knowledge and knowledge about how to direct the eye and the hands. Within ACT-R, all rules have an associated strength. When several rules apply to the same task situation (i.e., there are several rules whose conditions are satisfied), the rule with the highest strength is selected. The strength of rules is altered on the basis of their success and failure. If a rule helped generate a construction the model believes to be correct, then the rule's strength is increased. If a rule helped generate a construction the model believes to be incorrect, then the rule's strength is decreased. The manipulation of the strength of rules means that as the task progresses, the strategies (i.e., rules) that result in success are increasingly likely to be used.

The model begins with general rules for fitting blocks together. When the model has produced a construction that it believes to be a success, a new rule that describes the specific block features that were fit together is learned. The rule is therefore more specific than the previous, general, rule. As the task progresses, the number of rules that can be applied to fit blocks increases, and therefore more emphasis is placed on the mechanism for learning success and failure.

The representations of the blocks and block features in the model all have associated activations. When the memory elements for the blocks and block features are created, they are set to a base-level activation. Activation is then subject to decay on each cycle (a cycle involves the selection and firing of a single rule), but some blocks and block features can also have their activation raised based on the current goals of the model. When the activation of blocks and block features falls below a specified level (the retrieval threshold), they can no longer be matched in rule conditions. This decay represents a capacity limitation within the model.

Comparison of the Model With Adult Data

Adult subjects ($N = 5$) who had never encountered the task before were shown a picture of the completed tower. The subjects were then presented with the 21 blocks and asked to

Table 1. *An example rule from the ACT-R Tower model*

```
(P grab-constrn-left-hand-not-over-constrn
  ;; IF the goal is to grab a construction using the left hand
  =goal>
    ISA             grab-constrn
    constrn         =constrn
    hand            left
    goal-done       nil
  ;; AND the left hand is NOT currently near the construction
  ;; Note: "-" represents NOT
  =left-hand>
    ISA             left-hand
    -is-over        =constrn
  ;; AND the model's eye is looking at the construction so that
  ;; it knows where to move the left hand to
  =focus>
    ISA             focus
    object          =constrn
==>
  ;; THEN create a goal to move the left hand to the construction
  ;; and push the goal onto the goal stack
  =newgoal>
    ISA             move-hand
    constrn         =constrn
    hand            left
  !push!            =newgoal
)
```

Note. This rule moves the left hand to a construction. The rule contains three elements in its condition side and one element in its action side.

construct the tower, once only and unaided, while giving verbal protocols. Construction behavior was video recorded, together with onset timings to the nearest second.

For the model's timing data, the ACT-R default timing of 50 ms per rule firing was used. This timing was consistent for rules involving eye movements (saccades), but was increased to 200 ms for rules involving fixations, based on a summary of the vision literature provided by Baxter and Ritter (1996). For rules involving motor actions (fitting, rotating, moving, and disassembling blocks), the timing was increased to 550 ms (Jones & Ritter, 1997). Thus, the time predictions made by the model were absolute predictions; they were not scaled or otherwise adjusted in the analyses and graphs presented here.

The adult model began with the initial knowledge of the task that subjects had, such as that blocks of the same size go together, pegs go in holes, intended constructions are flush on their outer edges, and so on. The initial knowledge was apparent in the construction behavior of adults (e.g., they produced a total of 33 peg-pair or hole-pair constructions and 13 two-peg or two-hole constructions, and never considered producing constructions that were not flush on their outer edges).

The ACT-R expected-gain-noise parameter adds a random amount of noise to the strength of rules. It was set to 0.04 to randomize the selection of rules having the same strength. The usual settings range from 0.0 (the default) to 1.0 (J.R. Anderson, personal communication, February 4, 1999). The initial activation level given to memory elements also differed from the default, because of the decay mechanism (see Jones & Ritter, 1998, for an explanation). All other ACT-R parameters were either unused or set to the suggested default setting.

In addition to expected gain noise, the model contains other random elements (e.g., when the model is searching for blocks, an unseen block in the periphery is randomly selected). To achieve a better approximation to the expected model values, we compared the aggregate data from 10 runs of the model with the performance of the 5 adult subjects.

There are at least nine measures of behavior that can be taken for the Tower task, and each can be taken as an overall measure and on a layer-by-layer basis. Overall measures provide a guide to performance on the task (e.g., the time taken to complete the tower), and layer-by-layer measures show any learning that occurs (e.g., the time to produce each layer). The main measures reported here are the time taken to complete the tower and the number of constructions made in completing it (the number of constructions made refers to the raw number of correct and incorrect constructions produced while completing the tower).

For overall measures, the model provided a close match to the adult subjects' data for both the time taken to complete the tower and the number of constructions made, as shown in

Table 2. *Time taken and number of constructions made in completing the tower*

Measure	Adults (N = 5)	Models (N = 10)				Children (N = 5)
		Adult	RT6	9Cond	EGN6	
Time taken (in seconds)	126.6	129.0	160.2	188.2	166.3	174.0
	(34.0)	(31.5)	(35.6)	(41.5)	(37.9)	(32.1)
Number of constructions made	22.8	23.1	25.8	29.6	25.1	27.6
	(2.9)	(2.4)	(3.7)	(4.9)	(2.9)	(3.5)

Note. Standard deviations are shown in parentheses.

Table 2. The model actually provided a close match to the adult subjects' data on seven of nine measures of behavior (Jones, 1998), but only the two primary measures are reported here.

Performance on the Tower task is expected to show learning because each layer has the same characteristics, so problem solving on subsequent layers should become faster. The effect of learning on task performance can be examined from the time taken and the number of constructions made on a layer-by-layer basis. Figure 3 shows the mean time taken for the model (N = 10) and adult subjects (N = 5) to produce each layer and the mean number of constructions made in producing each layer. There was a good correlation between the model and the adult subjects for the mean time to produce each layer (r = .96; root mean square error [*RMSE*] = 4.1%), but not for the mean number of constructions made in producing each layer (r = .40; *RMSE* = 5.7%) because the curve for adults was flat. The RMSE was good for both measures.

The model's times and construction attempts did not decline after the size4 layer (see Fig. 3). (Block sizes are labeled in size order, with size6 referring to the largest blocks in the task, and size1 to the smallest, pinnacle block.) For the adults, the measures were at their lowest for the size3 layer. These results suggest that the model was performing at its optimum level after it completed the size4 layer, whereas adults were performing at their optimum level after completing the size3 layer.

MODIFYING THE ADULT MODEL

The adult model, because it provided a good match to the adult behavior, could be modified to incorporate proposed theoretical mechanisms of development. We compared the ensuing behavior with the behavior of 7-year-old children on the Tower task to test theoretical mechanisms of development.

Three independent modifications, two implementing capacity mechanisms and one implementing strategy choice, were made to the adult model. The results of each modification were compared against the performance of 7-year-old children, taken from a recent study by Reichgelt, Shadbolt, Paskiewicz, Wood, and Wood (1993). The children were tutored in completing the Tower task and then asked to build the tower unassisted. The posttest performance of *contingently tutored* (see Wood & Middleton, 1975) children (N = 5) is reanalyzed here. That is, the children's data reported here come from the second time the children completed the tower, when they were unaided. Using tutored children is not considered problematic because the study from which the data were taken indicated that contingent tutoring is not needed from the age of 6 and upward (Reichgelt et al., 1993). The same comparisons made between the original model and adults' performance were made between the modified model and the performance of the 7-year-old children.

Limited-Capacity Models

There are several capacity-based theories of development, and each explains capacity in a different way. For Pascual-Leone (1969), capacity is the number of knowledge structures that can be active at any one time. For Case (1985), capacity remains invariant with age, but through experience can be used more efficiently (e.g., chunking knowledge means the same

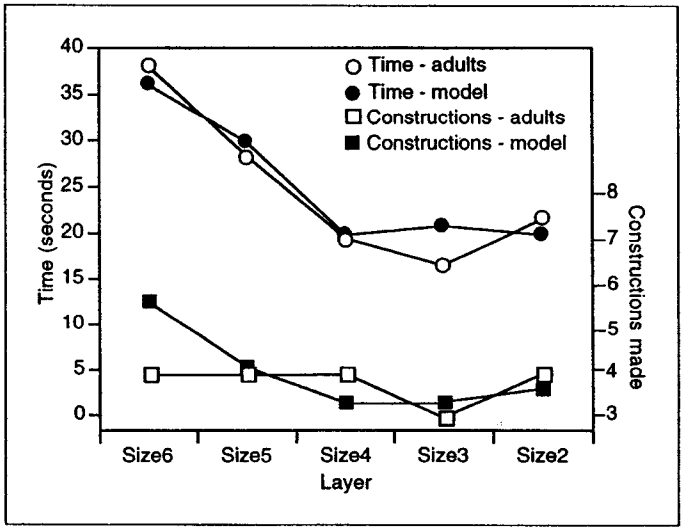

Fig. 3. Time taken and number of constructions made in completing each of the five layers of the Tower task, for the adults and the model (size6 = largest layer of blocks, size2 = smallest layer of blocks; the size1 pinnacle block is omitted because it is trivial).

knowledge can be represented using less capacity). Increased efficiency allows more items to be processed simultaneously. For Halford (1993), capacity is measured in terms of number of dimensions, which increases with age; at any given stage, only knowledge structures of a specific complexity can be represented.

The ACT-R cognitive architecture provides two natural ways to limit capacity in the adult model. First, the activation of blocks and block features must be above a retrieval threshold, so increasing retrieval threshold restricts the number of blocks and block features in working memory. Limiting elements in working memory corresponds to Pascual-Leone's (1969) theory that a limited number of elements can be active at any one time. Second, limiting the number of elements that a rule can have in its condition may mean that some strategies cannot be used, or have to be implemented as more than one rule (if possible). Limiting the number of elements a rule can have corresponds to Case's (1985) and Halford's (1993) theories that there is a limit to the number of elements that can be processed simultaneously.

Increasing Retrieval Threshold

The capacity mechanism described by Pascual-Leone (1969) was operationalized by increasing the retrieval-threshold parameter in the adult model. In ACT-R, memory elements need to be above the retrieval threshold in order to be considered active, so increasing the parameter reduces the number of active memory elements. This means that fewer memory elements can be considered for matching in rules. In the adult model, the retrieval threshold was set to the ACT-R default of 0 (the parameter usually ranges from 0 to 3 for adults; C. Lebiere, personal communication, February 4, 1999). We modified the model by varying the retrieval-threshold parameter from 1 to 9 in units of 1 (the initial level of activation of items in memory is 10.0, but items are subject to decay on each cycle). The model provided the best match to the children when the retrieval threshold was 6 (henceforth referred to as the RT6 model).

Table 2 shows that the RT6 model modified the adult model to be more childlike in terms of the time taken to complete the tower and the number of constructions made in completing the tower. However, for these same measures, the behavior of the RT6 model and the children did not correlate well on a layer-by-layer basis. The correlation between the model and the children's performance was $r = .51$ (*RMSE* = 27.6%) for the mean time to produce each layer and $r = .53$ (*RMSE* = 26.8%) for the mean number of constructions made in producing each layer.

The modification of retrieval threshold shows that limiting the number of memory elements that can be active at any one time can shift the behavior of the adult model toward that of 7-year-old children. However, the layer-by-layer data show that the RT6 model failed to match the layer-by-layer scores of the children.

Modifying the Number of Memory Elements in the Condition of a Rule

The developmental theories described by Case (1985) and Halford (1993) both specify a limit on the number of elements that can be processed simultaneously. In the adult model, the number of elements that can be processed simultaneously corresponds to the number of memory elements specified in the condition of a rule. We operationalized the capacity theories of Case and Halford by limiting this number.

The number of memory elements in the conditions of rules varies from 1 to 12 in the adult model. There are three methods by which the number of memory elements can be reduced (should the number of elements in a rule exceed a specified limit): deleting the rule, splitting the rule into two or more sequential rules, and chunking memory elements so that the same information can be represented using fewer elements. We used the first two methods to examine the effect of limiting the number of memory elements in the condition of a rule (the model does not implement the third method as yet).

The limit of memory elements in the condition of a rule was varied from 7 to 10. To implement the highest level (10), we deleted 3 rules and split 4 additional rules in two. To implement the lowest level (7), we deleted 3 rules (the same ones as for the 10-element limit) and split 16 others in two. Rules were deleted only when they could not be broken down into smaller rules. The model provided the best match to the children's behavior when the limit on elements in the conditions of rules was 9 (henceforth referred to as the 9Cond model).

Table 2 shows that the 9Cond modification enabled the model's behavior to closely resemble the children's behavior for both time taken to complete the tower and number of constructions made in completing the tower. However, when these measures were examined on a layer-by-layer basis, the behavior of the 9Cond model and the children did not correlate well for either the mean time to produce each layer ($r = .37$; *RMSE* = 34.7%) or the mean number of constructions made in producing each layer ($r = .44$; *RMSE* = 31.2%). As with the RT6 model, the overall measures showed a good match between the model and the children's behavior, but the layer-by-layer measures did not.

Strategy-Choice Model

Siegler and Shipley (1995) suggested that children's strategy use changes with experience. This idea implies that older children are more able to select appropriate strategies than their younger counterparts. Strategy choice in children's addition has been examined in models before (Shrager & Siegler, 1998; Siegler & Shrager, 1984), and our examination of strategy

choice therefore represents a test of its applicability across domains.

The adult model implements strategies as rules, or sequences of rules. When more than one rule can be applied, the one with the highest strength is selected. The model subjects the strength of rules to an expected gain noise of 0.04, which means that on some occasions, rules that do not have the highest strength are selected. The noise is placed on every rule in the model, thereby influencing strategy choice for all of the various strategies involved in constructing the tower. Increasing the value of the noise reduces the model's knowledge about which rules are the most effective, making it possible to examine the model's behavior when strategy choice is not as effective as it is for the adult model.

The expected gain noise placed on the strength of rules was increased in steps of 1 unit, starting at 1. The range of values for the parameter normally lies between 0 and 1 for adults, meaning the maximum setting can be open-ended for children. The model provided the best match to the behavior of the 7-year-old children at an expected-gain-noise level of 6 (henceforth this is referred to as the EGN6 model).

Table 2 shows that the expected-gain-noise modification enabled the behavior of the model to move closer to that of the children for both time taken and number of constructions made. Figure 4 shows the mean time to produce each layer and the mean number of constructions made in producing each layer for both the EGN6 model and the children. There was a good correlation between the model and the children for both measures (r = .99 and *RMSE* = 3.0% for time taken to construct each layer; r = .73 and *RMSE* = 26.8% for constructions made per layer). The modification enabled a good fit on layer-by-layer measures.

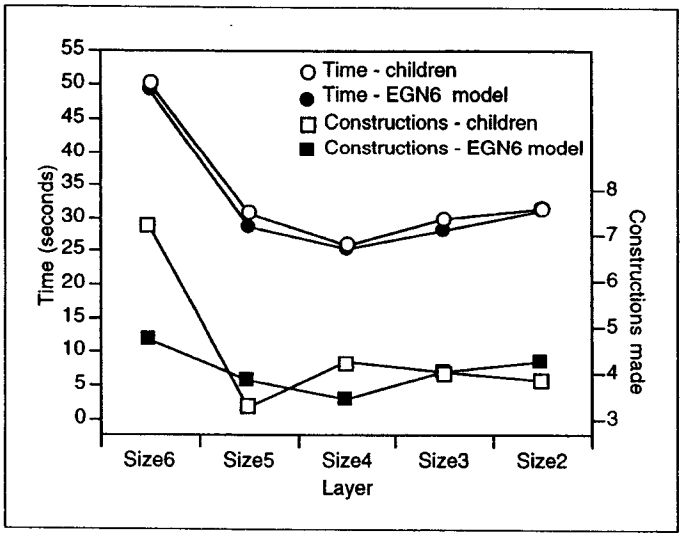

Fig. 4. Time taken and number of constructions made in completing each of the five layers of the Tower task, for the 7-year-old children and the EGN6 model (size6 = largest layer of blocks, size2 = smallest layer of blocks; the size1 pinnacle block is omitted because it is trivial). See the text for an explanation of the model.

An interesting characteristic that the EGN6 modification provided was an increase in time to produce the size3 and size2 layers, an increase that matches the children's behavior. By the time the EGN6 model begins the size3 layer, some aspects of the size3 and size2 blocks are not known (during the task, decay has reduced their activation to below the retrieval threshold). The likelihood of an incorrect strategy being selected is therefore increased because there is less opportunity for meeting the conditions of rules that represent good strategies. Decay of blocks and block features is not as problematic for the adult model because it takes less time to complete the first three layers.

The overall fit between the behavior of the EGN6 model and the 7-year-old children was very good. There were no reliable differences on any of the individual layer measures, for both timing and construction attempts. Modifying strategy choice provided a good fit to the 7-year-old children's data.

SUMMARY

Independent, theoretically motivated modifications to a computational model of adult performance (implemented within a cognitive architecture) enabled the model to approximate the behavior of 7-year-old children on the Tower task. Each modification represented an interpretation of a theoretical mechanism of development, changing the behavior of the model in a different way. Some modifications matched the data from 7-year-olds better than others. Changes to strategy choice provided the best fit to the children's data.

There is widespread belief that furthering the understanding of cognitive change is of critical importance to the progress of developmental theory (Siegler, 1989; Sternberg, 1984). This investigation of the role of different proposed mechanisms of development has illuminated the influence that each mechanism has on behavior (albeit in a single domain).

The model and task used have also shown that subject and model data can be compared on multiple measures within a developmental model, including timing data. Previous computational models of development were compared with subject data on one or two measures (e.g., McClelland & Jenkins, 1991), and rarely included timing data. Comparing model and subject data on a variety of measures provides a more detailed match, warrants more confidence that the processes modeled were correct, and suggests further ways the model can be improved.

With additional work, models of other ages and other tasks can be created. These models will be able to explain individual differences within age groups, as well as the progression between ages (in terms of differences between the models rather than transition mechanisms, for the moment). In both cases, the models should be able to highlight the knowledge differences or architectural changes that lead to the differences in behavior.

The mechanisms that are shown to influence the behavior of the models will then become the focus for any transition mechanisms, because transition mechanisms must account for them. In this way, the degrees of freedom for transition mechanisms can be reduced (Simon & Halford, 1995).

The next step is to carry out further independent modifications of the model, and then examine interactions by including two or more modifications within the same model. The effects of interacting modifications are expected to be stronger than the effects of individual modifications, enabling the model to match the 7-year-old children's behavior better than the individual modifications. This type of work will help to more directly answer the question, "What develops?"

Acknowledgments—The authors would like to thank Peter Cheng and Robert Siegler for useful comments during the preparation of this article, Peter Cheng and Heather Wood for creating the initial version of Figure 1, and Hans Reichgelt, Nigel Shadbolt, Theresa Paskiewicz, and Heather Wood for collecting the videotaped data of 7-year-old children completing the Tower task.

REFERENCES

Anderson, J.R. (1993). *Rules of the mind.* Hillsdale, NJ: Erlbaum.

Baxter, G.D., & Ritter, F.E. (1996). *Designing abstract visual perceptual and motor action capabilities for use by cognitive models* (Technical Report No. 36). Nottingham, England: University of Nottingham, Centre for Research in Development, Instruction and Training.

Case, R. (1985). *Intellectual development: Birth to adulthood.* Orlando, FL: Academic Press.

Elman, J.L., Bates, E.A., Johnson, M.H., Karmiloff-Smith, A., Parisi, D., & Plunkett, K. (1996). *Rethinking innateness: A connectionist perspective on development.* Cambridge, MA: MIT Press.

Gentner, D., Rattermann, M.J., Markman, A., & Kotovsky, L. (1995). Two forces in the development of relational similarity. In T.J. Simon & G.S. Halford (Eds.), *Developing cognitive competence: New approaches to process modeling* (pp. 263–313). Hillsdale, NJ: Erlbaum.

Halford, G.S. (1993). *Children's understanding: The development of mental models.* Hillsdale, NJ: Erlbaum.

Jones, G. (1998). *Testing mechanisms of development within a computational framework.* Unpublished doctoral dissertation, University of Nottingham, Nottingham, England.

Jones, G., & Ritter, F.E. (1997). Modelling transitions in children's development by starting with adults. In *Proceedings of the European Conference on Cognitive Science (ECCS '97)* (pp. 62–67). Manchester, England: Society for the Study of Artificial Intelligence and the Simulation of Behaviour.

Jones, G., & Ritter, F.E. (1998). Initial explorations of modifying architectures to simulate cognitive and perceptual development. In *Proceedings of the Second European Conference on Cognitive Modelling* (pp. 44–51). Nottingham, England: Nottingham University Press.

Kail, R. (1991). Development of processing speed in childhood and adolescence. In H.W. Reese (Ed.), *Advances in child development and behavior* (Vol. 23, pp. 151–185). London: Academic Press.

Klahr, D., & Siegler, R.S. (1978). The representation of children's knowledge. In H.W. Reese & L.P. Lipsett (Eds.), *Advances in child development and behavior* (Vol. 12, pp. 61–116). New York: Academic Press.

Klahr, D., & Wallace, J.G. (1976). *Cognitive development: An information processing view.* Hillsdale, NJ: Erlbaum.

McClelland, J.L., & Jenkins, E. (1991). Nature, nurture and connections: Implications of connectionist models for cognitive development. In K. VanLehn (Ed.), *Architectures for intelligence* (pp. 41–73). Hillsdale, NJ: Erlbaum.

Murphy, C.M., & Wood, D.J. (1981). Learning from pictures: The use of pictorial information by young children. *Journal of Experimental Child Psychology, 32,* 279–297.

Murphy, C.M., & Wood, D.J. (1982). Learning through media: A comparison of 4–8 year old children's responses to filmed and pictorial instruction. *International Journal of Behavioural Development, 5,* 195–216.

Newell, A. (1990). *Unified theories of cognition.* Cambridge, MA: Harvard University Press.

Pascual-Leone, J. (1969). *Cognitive development and cognitive style.* Unpublished doctoral dissertation, University of Geneva, Geneva, Switzerland.

Piaget, J. (1950). *The psychology of intelligence* (M. Piercy & D.E. Berlyne, Trans.). London: Routledge & Kegan Paul. (Original work published 1947)

Reichgelt, H., Shadbolt, N.R., Paskiewicz, T., Wood, D.J., & Wood, H. (1993). EXPLAIN: On implementing more effective tutoring systems. In A. Sloman, D. Hogg, G. Humphreys, D. Partridge, & A. Ramsay (Eds.), *Prospects for artificial intelligence* (pp. 239–249). Amsterdam: IOS Press.

Ritter, F.E., Baxter, G.D., Jones, G., & Young, R.M. (in press). Supporting cognitive models as users. *ACM Transactions on Computer-Human Interaction.*

Shrager, J., & Siegler, R.S. (1998). SCADS: A model of children's strategy choices and strategy discoveries. *Psychological Science, 9,* 405–410.

Shultz, T.R., Schmidt, W.C., Buckingham, D., & Mareschal, D. (1995). Modeling cognitive development with a generative connectionist algorithm. In T.J. Simon & G.S. Halford (Eds.), *Developing cognitive competence: New approaches to process modeling* (pp. 205–261). Hillsdale, NJ: Erlbaum.

Siegler, R.S. (1989). Mechanisms of cognitive development. *Annual Review of Psychology, 40,* 353–379.

Siegler, R.S. (1995). Children's thinking: How does change occur? In W. Schneider & F. Weinert (Eds.), *Memory, performance and competencies: Issues in growth and development* (pp. 405–430). Hillsdale, NJ: Erlbaum.

Siegler, R.S., & Jenkins, E.A. (1989). *How children discover new strategies.* Hillsdale, NJ: Erlbaum.

Siegler, R.S., & Shipley, C. (1995). Variation, selection and cognitive change. In T.J. Simon & G.S. Halford (Eds.), *Developing cognitive competence: New approaches to process modeling* (pp. 31–76). Hillsdale, NJ: Erlbaum.

Siegler, R.S., & Shrager, J. (1984). Strategy choices in addition and subtraction: How do children know what to do? In C. Sophian (Ed.), *Origins of cognitive skills* (pp. 229–293). Hillsdale, NJ: Erlbaum.

Simon, T.J., & Halford, G.S. (1995). Computational models and cognitive change. In T.J. Simon & G.S. Halford (Eds.), *Developing cognitive competence: New approaches to process modeling* (pp. 1–30). Hillsdale, NJ: Erlbaum.

Sternberg, R.J. (1984). Mechanisms of cognitive development: A componential approach. In R.J. Sternberg (Ed.), *Mechanisms of cognitive development* (pp. 163–186). Prospect Heights, IL: Waveland Press.

Wood, D., & Middleton, D. (1975). A study of assisted problem solving. *British Journal of Psychology, 66,* 181–191.

Wood, D.J., Bruner, J.S., & Ross, G. (1976). The role of tutoring in problem solving. *Journal of Child Psychology and Psychiatry, 17,* 89–100.

Young, R.M. (1976). *Seriation by children: An artificial intelligence analysis of a Piagetian task.* Basel, Switzerland: Birkhauser.

(RECEIVED 3/16/99; ACCEPTED 8/31/99)