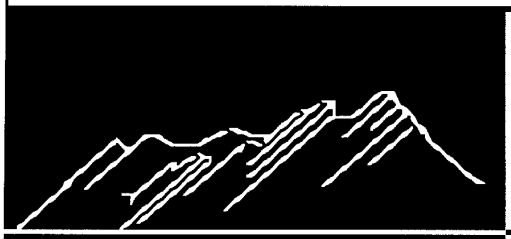
Institute of Cognitive Science



Technical Report

University of Colorado, Boulder

The Construction-Integration Model: A Program and Manual

Ernest Mross and Jonathan Roberts

Institute of Cognitive Science
University of Colorado at Boulder
Boulder, CO 80309-0344

The Construction-Integration Model: A Program and Manual

Ernest Mross and Jonathan Roberts

Institute of Cognitive Science
University of Colorado
Boulder, CO 80309-0345

Tech Report #92-14

Table of Contents

The CI Model Program

Welcome to the CI Model program. "CI" stands for "construction-integration," after Kintsch's (1988) model of discourse comprehension. The program was written in an effort to simplify the application of Kintsch's model to various phenomena in discourse comprehension research. This document represents the instructions for how to use the program. As such, it assumes a degree of familiarity with the model. It is not intended as a substitute for that familiarity. The user should be well-versed with the original description of the model, as well as subsequent papers describing any modifications in the original model (e.g., Kintsch, 1992a; Kintsch, 1992b; Kintsch, 1991; Kintsch, 1988; Kintsch & Welsch, in press; Kintsch, Welsch, Schmalhofer, & Zimny, 1990; Mannes & Kintsch, 1991; Otero & Kintsch, in press).

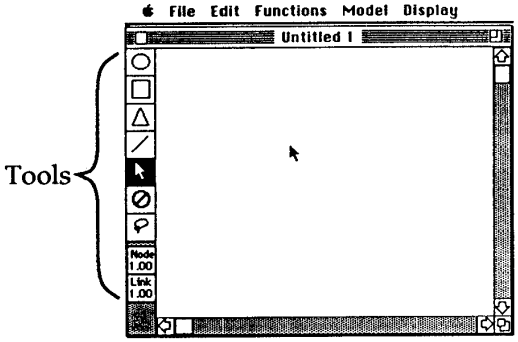
Briefly, the program allows the user to create a graphical network of nodes (consisting of text propositions, associations to those propositions, and the like) along with the links among those nodes. From this, the program constructs a matrix (the "coherence matrix" in the model) of the information held in the graphical network. In addition, the program constructs a vector corresponding to the activation values of the nodes in the network. As dictated by the CI model, the activation vector is repeatedly postmultiplied by the coherence matrix. After each postmultiplication (or iteration), the activation vector is normalized. The difference between old and new activation values is calculated, and the iteration process is stopped after some arbitrary criterion is reached (such as a mean change in activation value less than .001).

The program's graphical interface, similar in flavor to that of MacDraw©, was designed to allow for the easy creation and manipulation of nodes and links. The program provides a set of "Tools" with which to draw nodes and links, position nodes on the screen, and otherwise manage the properties of the network being created. Besides the tools, there are menu-based items that serve to do everything from controlling what information appears on the screen to performing the matrix multiplication. Before going on to a description of all the capabilities of the program, there is one particular concept to be understood that will enhance the modeler's use of the program. This is the idea of "Active" versus "Inactive" nodes. This concept has nothing to do with the model, per se, and everything to do with the program. By convention, model calculations (i.e., integration) in the program are only performed on active nodes. Inactive nodes do not take part in computations. For more information about this distinction, see the description under the "Pointer Tool" and the menu commands "Activate selected nodes" and "Deactivate selected nodes" under the **Functions** menu.

First, the Tools will be described, followed by the various Menus associated with the program. Figures of the discussed material have been included where helpful for better reference. Finally, the last section of this manual contains a tutorial. It describes in step-by-step fashion some of the basic operations possible with the CI Model program. Go through the tutorial to familiarize yourself with how the model works.

Tools

The Tools are used for creating and deleting nodes and links, positioning nodes, and otherwise overseeing the creation of the network. To choose a particular tool, simply click once in the appropriate box. The chosen tool will then become darkened. Each tool (or set of tools) will be described in the order in which it appears from top to bottom in the adjacent figure.



Node Tools 

The node tools consist of the first three tools in the above figure (i.e., the circle, square, and triangle). These tools are used for creating nodes. Once a node tool has been chosen, a node is created by clicking in the model window. More than one node tool is provided in case it is useful for the user to distinguish between/among different types of nodes, for example, propositional nodes and situation model nodes. The circle tool could be used for propositional nodes and the square tool could be used for situation model nodes. The choice is merely for convenience.

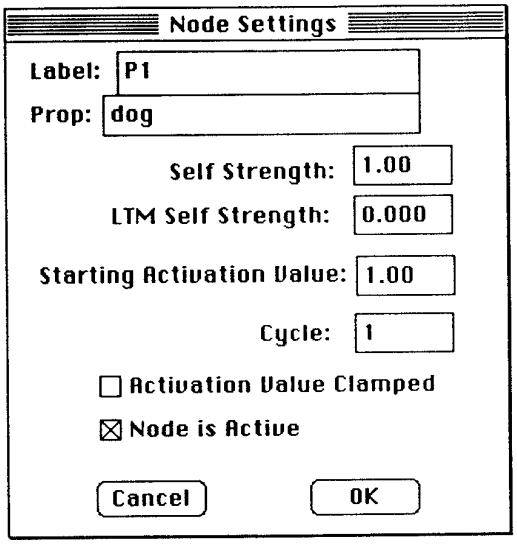
Link Tool 

The link tool is used to create links between nodes. When the link tool is chosen, the cursor becomes a cross hairs. To create a link, position the cross hairs on an existing node. Next, click and hold down the mouse and drag the cursor to the node to which it is to be linked. When the cross hairs are on the second node, release the mouse. A link will then be drawn between the two nodes.

In addition, the link tool can be used to modify or delete existing links. For example, say the user wishes to increase the strength of a previously drawn link. To do this, the user chooses the link tool and redraws (i.e., draw the link again as if it didn't exist) the link that is to be modified This will bring up a dialog box which presents several options. The first option that appears at the top of the dialog box is "Link Strength." The current link strength is displayed in the box, but this value can be selected and replaced. The second option of the dialog box is for modifying the LTM link strength. Clicking the "OK" button at this point causes any changes to take effect, whereas clicking "Cancel" causes no changes to occur. The other two options presented in this dialog box are "Click here to delete this link!", which should be self-explanatory, and "Click here for bidirectional link." Clicking this button causes another dialog box to come up which allows the user to modify either the WM link strength or the LTM link strength in such a way that the strength of the link from node A to node B is different from the strength of the link from node B to node A.

Pointer Tool 

The pointer tool is used for selecting nodes and also for moving nodes around on the screen. There are two ways to select nodes using this tool. One way is to choose the pointer tool and then click and drag down and to the right from the initial click. This will create the outline of a rectangle as you drag. Any nodes that are contained within this rectangle will be selected (i.e., darkened) after you release the mouse. The other way to select nodes is by "shift-clicking" them. That is, the user can always add nodes to the current selection by holding down the shift key and clicking on additional nodes.



The pointer tool can also be used to modify several characteristics of a node. Double-clicking a node with the pointer tool brings up a dialog box that gives the user access to these characteristics. The dialog box (labeled "Node Settings") that appears is depicted in the adjacent figure. Most of the properties of the node that are modifiable should be self-explanatory. For example, the node's label, its proposition, its self strength (this corresponds to the node's diagonal element in the coherence or weight matrix), its LTM self strength, and its starting activation value (in the initial activation vector). "Cycle" corresponds to an integer that can be associated with a node and represents the sentence or cycle (e.g., Miller & Kintsch, 1980) that a proposition is from. The program doesn't necessarily treat propositions from different cycles differently; the cycle number is more an aid to the modeler for keeping track of multi-sentence inputs. (For more information about cycle number, see "File" under the **Menus** heading and read about the Open command.)

The penultimate characteristic that is modifiable via this dialog box is "Activation Value Clamped." "Activation value clamped" means that this node's activation value will not change from iteration to iteration. The node will retain its starting activation value throughout, and thus its influence on its neighbors will remain constant as well. An example of why one would want to clamp a node could be to model the effects of a stimulus that is always present or available to the subject, such as a set of instructions or the like. Consult the relevant publications for more information.

The final characteristic that is modifiable is "Node is Active." This property has far less to do with the construction-integration model than it has to do with how the program is used. When using of the program, it is important to distinguish between "active" and "inactive" nodes. Active nodes are those that take part in model calculations (i.e., integration). Inactive nodes do not take part in calculations. This is so that the user can read in a text of several sentences (or, more precisely, the propositions

representing those sentences) with the program and then model the text cycle by cycle (or sentence by sentence). Initially all propositions are activated, but the user can "select" a subset of propositions to be deactivated. The user can make a node inactive (or active, depending on its current state) by clicking in the box next to "Node is Active." There is a method for activating/deactivating more than one node at once. See "Activate selected nodes" and "Deactivate selected nodes" under the **Functions** menu.

Delete Node Tool 

The function of this tool is self-explanatory. Choosing this tool and clicking on a node will bring up a dialog box asking the user to confirm that the clicked-on node should be deleted. This is not a reversible action.

Lasso Tool 

The lasso tool is for selecting nodes that are not adjacent to each other. After choosing this tool, the user can click and drag around the screen in any manner desired as with a pencil on paper. Any nodes within the circumscribed area become selected when the mouse button is released, and the tool becomes the pointer tool.

Node Strength Box 

The node strength box is used for changing the default value used for newly created nodes. A single click in this box brings up a dialog box that asks the user to enter a new default node strength in the appropriate place. Clicking "OK" causes the entered strength to become the new default; clicking "Cancel" causes no changes to occur. The default value upon program startup is 1.00.

Link Strength Box 

The link strength box is used for changing the default value used for newly created links. It works identically to the node strength box. The default value upon program startup is 1.00.

Menus

File

The File menu's appearance is similar to that of most other Macintosh applications, with New, Open, Close, Save, etc. With the Open menu option it is possible to open two different kinds of files. One kind of file is called a "model" file. This type of file is created when the user chooses "Save" or "Save As..." from the file menu. When a file of this type is opened, it appears on the screen as a graphical representation of the nodes and links in the network. The other kind of file that can be

opened is a simple text or ASCII file containing the propositions (i.e., the nodes) that make up the network, plus some other information.

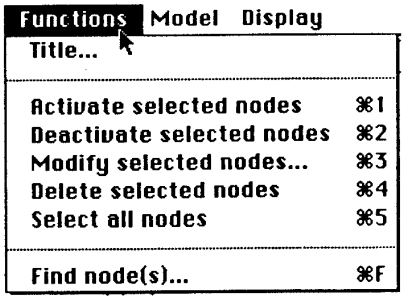
It is possible to have the program read in a list of propositions and create a network. This network will contain many of the desired links between propositions, based on argument overlap and propositional embedding, after the work of Kintsch and van Dijk (1978; van Dijk & Kintsch, 1983). When "Open" is chosen from the File menu, a dialog box appears that lists all "Model" files (i.e., files previously created with the program) and any text files residing in the current folder. If a text file is chosen, the program brings up a dialog box giving the user some options on what to do with the input file. The first option is "Automatic link generation." Choosing this option indicates that the user would like the program to calculate the links among propositions in the file based on argument overlap and propositional embedding. Choosing this option also enables the option "Base link strength on number of arguments shared." This option is not what most users are going to want, and so it will not be described here further. The final available option is choosing the default link strength to be used for any links. The default link strength is 1.0. Of course, if automatic link generation is not chosen this option has no effect. Clicking "OK" causes the program to read the input file and calculate any potential links based on the previously described criteria. The program has a crude algorithm for positioning in the window the nodes it creates. The user will likely wish to move groups of nodes around for a more readable and aesthetic look.

The format of the propositional input file will be described briefly. An example file should have come with a copy of the CI model program. Adhere strictly to this format! The program is intolerant of variations. An incorrect format can cause the program to calculate links incorrectly, at best, and crash the program, at worst. The format is very simple, and the input file can be created with any word processing program that allows you to save a document in text format. The first line of the input file contains an integer corresponding to how many propositions (or nodes-to-be) are listed in the file. Each subsequent line contains three items: A number corresponding to the cycle which the proposition is from, a proposition label (e.g., "P1"), and the proposition itself (e.g., "Threw[Chuck,Debbi,ball]"). One important thing to note is that there should only be a single space between each of the three entries on a line. Another important point is that arguments to propositions should be enclosed in square brackets (as opposed to parentheses). Finally, it is useful to know that the program will calculate argument overlap and propositional embedding based on either the proposition label or the proposition name. Thus, as in the example file, arguments of propositions can be of the form "P*n*" or "dog," for example.

Edit

The Edit menu appears in the list of menus for the CI model program, but the only menu option currently working is "Show Clipboard."

Functions



In general, the Functions menu is used to manipulate various characteristics of nodes and links created with the model. Two other useful functions can be found here as well. Each function will be described as it appears from top to bottom under the Functions menu.

Title.... This menu item allows you to specify a title that will appear at the top center of the model window. The title will be saved with the document.

Activate selected nodes. This menu item adds the currently selected nodes to the active node list. Computations (i.e., integration) are only performed on active nodes. See the information under the Pointer and Lasso tools for methods on how to select nodes.

Deactivate selected nodes. This menu item removes the currently selected nodes from the active node list. Thus, it is opposite in effect to the preceding "Activate Selected Nodes."

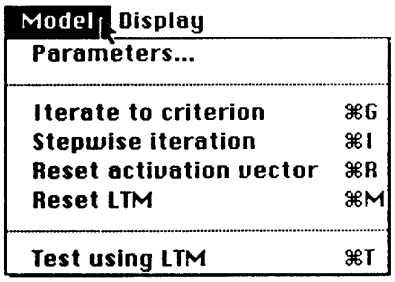
Modify selected nodes.... This menu item brings up a dialog box which allows the user to modify various properties of the selected nodes. Those properties that are currently modifiable are the self strength, the link strength, the starting activation value, and the cycle number of the selected nodes. For example, if you wanted a particular node to have a stronger influence on its neighbors relative to other nodes, you could select this node, choose this menu item, and then type "2" (or whatever) into the box next to "Link Strength" and then click "OK." All links to other nodes from the selected node would then change to have a link strength of 2.

Delete selected nodes. This menu item is self-explanatory. A dialog box appears when it is chosen to make sure that the user intends to delete the selected nodes.

Select all nodes. This menu item is also self-explanatory. It allows the user to select all nodes in the model.

Find node(s).... This menu item allows the user to "find" nodes based upon certain search criteria. When it is chosen, a dialog box appears that allows the user to base the search on one of the following criteria: Activation value greater than some number (the default is 0.90), cycle number, or label. Any node fitting the chosen criterion is selected. The find node command is most useful in networks consisting of a large number of nodes where it is hard to keep track of how highly various nodes are activated.

Model



The Model menu contains the core functions used for performing the integration procedure of the CI model, plus a couple of other functions. As before, each will be described in turn from top to bottom as they appear under the Model menu.

Parameters.... This menu item contains access to some of the free parameters of the model. Currently, only the stopping criterion (default value = 0.001) is modifiable.

Iterate to criterion. This menu item postmultiplies the activation vector with the weight matrix until the stopping criterion is reached. Only active (i.e., non-grayed-out) nodes are used in the computation. After the criterion is reached, a dialog box is displayed asking whether the program should move the results to LTM. LTM is another matrix which is used to represent the strength of items in long-term memory. The mathematics of this procedure are described in detail elsewhere (Kintsch & Welsch, in press). The user is cautioned about moving results to LTM indiscriminately because the LTM calculation is additive. Thus choosing "iterate to criterion" several times in a row and answering "yes" each time when asked about moving results to LTM will result in an incorrect LTM representation. It is best to answer "no" until you are certain you wish to move on to the next processing cycle. When you are sure you want to move on, choose iterate to criterion one last time (for the current cycle) and respond yes when asked to move results to LTM. Then deactivate all nodes from that cycle not being carried over, and proceed in the same manner with the next cycle.

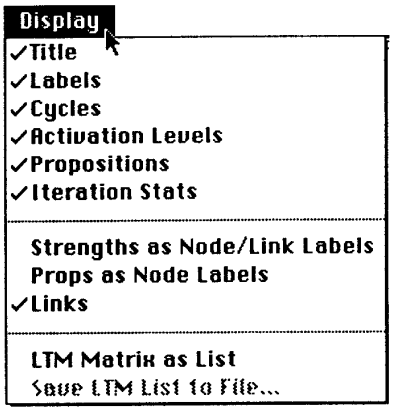
Stepwise iteration. This menu item postmultiplies the activation vector with the weight matrix a single time. Choosing it several times in succession allows the user to follow the time course of activation.

Reset activation vector. This menu item resets all active nodes to their starting activation values.

Reset LTM. This menu item resets all LTM matrix values back to 0.0. When this command is chosen a dialog box appears warning the user to make sure this is what is intended, as this command is not reversible.

Test using LTM. This menu item causes the integration to be performed using the LTM matrix instead of the working memory matrix. It also disables asking the user whether the results of the iteration should be moved to LTM when "iterate to criterion" is chosen (since it makes no sense in this context).

Display



The Display menu is used to control various display characteristics. Because these characteristics can be turned on or off, this is a checkmark menu. Those display features that are "on" at a given moment are checked. As usual, the menu options will be described from the top to the bottom of the menu.

Title. This menu item controls whether the title is displayed. If so, it appears centered and near the top of the window.

Labels. This menu item controls whether the proposition labels will appear in the list of node information on the left of the window. These labels typically take the form of "P1", "P2", etc.

Cycles. This menu item controls whether a given proposition's cycle will apear in the list of node information. Cycles appear in a c*n* format, where *n* is the cycle or sentence number which the proposition is from.

Activation Levels. This menu item controls whether the activation level of a given node appears in the node list information.

Propositions. This menu item controls whether the Proposition name (e.g., Threw[Chuck,ball]) appears in the node list information.

Iteration Stats. This menu item controls whether the results of the iteration process are displayed beneath the list of node information.

Strengths as Node/Link Labels. This menu item controls whether the self and link strengths of the network appear beneath each node and link of the graphical representation. Both WM and LTM strengths are represented. WM strengths appear above LTM strengths.

Props as Node Labels. This menu item controls whether the proposition names appear beneath each node of the graphical representation.

Links. This menu item controls whether existing links are drawn. For the majority of users, this item can be ignored. If, however, your network is extremely complex with hundreds to thousands of links, unchecking this item can greatly decrease the time it takes to redraw the screen. Use at your own risk.

LTM Matrix as List. This menu item controls whether the LTM matrix is displayed in list format. If this item is chosen, the display changes so that the graphical representation is replaced by a list representation. What is displayed is as follows: First

is the proposition label, followed by the cycle number, followed by the proposition name, followed by the LTM self strength for that proposition, followed by the proposition labels and LTM link strengths of any propositions that are connected to the first proposition.

Save LTM List to File.... This menu item allows the user to save the LTM list to a text file. Choosing this menu item brings up a dialog box for the user to enter a file name to which the matrix information is written. This menu option is only enabled if the previously described menu item, "LTM Matrix as List", is checked.

Tutorial

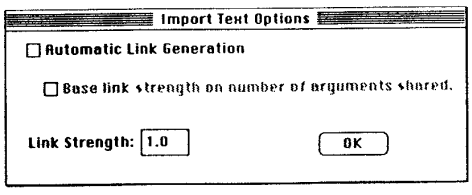
This section is a not-so-random walk through the CI Model program. In the context of a specific example, that of the "Tina and Lisa" camping text, the tutorial demonstrates some of the capabilities of the program. This is meant to be a "hands-on" example, so ideally the reader should start up the program and follow along, while performing the operations as they are described herein.

When preparing to model a particular text or set of texts, the first step should consist of propositionalizing the text(s). Bovair and Kieras (1985) have written a very readable introduction to propositional analysis, and this source should be consulted concerning questions of this type. For our example, the sentence "Tina gathered the kindling as Lisa set up the tent" was used. This sentence can be propositionalized into 7 propositions, which can be seen in the figure below. In addition, there are 3 knowledge elaboration propositions included as well, representing likely associations a typical reader might generate as they read this sentence. The knowledge elaborations consist of 2 propositions related to camping (i.e., "camping" and "fire") plus 1 proposition ("circus") related to the text proposition "tent." The text propositions and knowledge elaborations were entered into a Macintosh text file and that is what is displayed in the figure below. Nota Bene: If you use Microsoft Word to create the file with the propositions, you should **not** save the file in Word's default format. The file must be a text file. To create a text file using Word, click "Save As..." under the File menu, and then click the "File Format" button on the bottom lefthand of the dialog box. This brings up another dialog box, from which you should choose the option "Text Only with Line Breaks."

An example text input file is displayed in the adjacent figure. The first line of the file you create contains an integer corresponding to the number of propositions listed in this file. There are 10 propositions in our example. Thereafter follow 10 lines (for our example), with 3 items per line. The 3 items correspond to a cycle number for each proposition, a label (e.g., "P4"), and the proposition itself. Take care to follow the format of this file precisely, as the program is not tolerant of variation. For more information regarding the format of this file, please refer to the "File" subsection under the "Menus" heading of this document.

```
10
1 P1 TINA
1 P2 KINDLING
1 P3 GATHER[TINA,KINDLING]
1 A1 FIRE
1 A2 CAMPING
2 P4 AS[P3,P7]
2 P5 LISA
2 P6 TENT
2 P7 SET-UP[P5,P6]
2 A3 CIRCUS
```

The file containing the example propositions is entitled "Tina text example." Start up the CI Model program by double-clicking on one of the "CI for Mac..." application icons, the one appropriate for the computer you are using. Once the program has started,

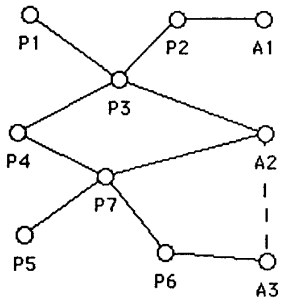


choose "Open" from the File menu. Then locate and double-click the "Tina text example" file. At this point, an "Import Text Options" (see the adjacent figure) dialog box should appear. By clicking in the "Automatic Link Generation" box and then clicking "OK", the user can direct the program to calculate links between nodes on the basis of argument overlap and propositional embedding. For more information about the options presented in this dialog box, refer to the "File" subsection under the "Menus" heading of this document.

The program uses a crude algorithm when initially positioning nodes read in from a text file. The reader should manipulate (by clicking and dragging) the nodes from their initial positions to something resembling the positioning in the figure below. In addition, the reader should draw in 5 more links. In the initial window, no links appear between any of the association (or knowledge elaboration) nodes and any other nodes. The modeler must add links between A1 and P2, A2 and P3, A2 and P7, and between A3 and P6. These links can be drawn by first clicking in the link strength box (at the bottom of the tools palette). This will bring up a dialog box which allows the user to change the default link strength with which new links are drawn. The default link strength (normally 1.0) should be changed to 0.50. Next, these new links can be drawn by selecting the link tool (see the reference section of this manual for a description of the link tool) and clicking and holding on a given node and drawing a line to another node and releasing the link tool. For the final link, the one between A2 and A3, a negative link should be drawn. A negative link can be drawn by drawing a link between these two nodes just as was done for the other nodes, but then in addition the modeler should redraw the link, tracing over the link just drawn. This will bring up the "Link Settings" dialog box which allows the user to enter the desired new link strength. Enter a "-" (minus) in front of the "1.00" that appears in the link strength box, and then click "OK." The link's appearance should then change to that of a dashed line, as in the figure, indicating that the link is negative in weight. For more information about the options in the "Link Settings" dialog box, refer to the Link Tool under the Tools section.

The modeler should take two further preparatory steps before the actual modeling can begin. Both of these steps concern the assocation nodes. These nodes' starting activation values (or values in the activation vector) need to be changed to 0 from their default value of 1.0. In addition, their self-strength values need to be changed to 0.5 from their default value of 1.0. The self strength is that value corresponding to the node's diagonal entry in the coherence matrix (or c_{jj}, where $i = j$). Association nodes' starting activation values are set to 0 because only text propositions are assumed to be activated initially. Their self strengths and link weights are set to 0.5 because it is assumed that they have less influence on the integration process than text propositions.

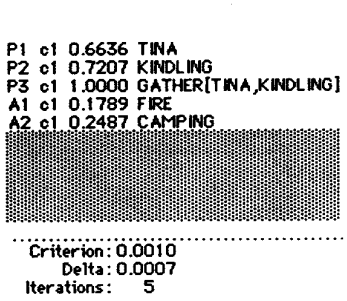
```
P1 c1 1.0000 TINA
P2 c1 1.0000 KINDLING
P3 c1 1.0000 GATHER[TINA,KINDLING]
A1 c1 0.0000 FIRE
A2 c1 0.0000 CAMPING
P4 c2 1.0000 AS[P3,P7]
P5 c2 1.0000 LISA
P6 c2 1.0000 TENT
P7 c2 1.0000 SET-UP[P5,P6]
A3 c2 0.0000 CIRCUS
```

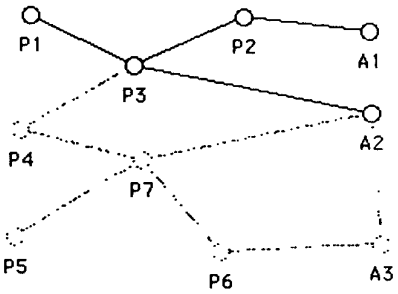


These two steps can be accomplished easily. First, the modeler should select the three association nodes. This can be done either by clicking and drawing a rectangle around the three nodes with the pointer tool (and then releasing the mouse), or by "shift-clicking", that is, holding down the shift key and clicking each node in turn. (Holding down the shift key instructs your Macintosh to add each object clicked to the current selection.) If the nodes have been properly selected, each node should appear darkened, and each node's corresponding line in the node list on the left of the window should appear darkened as well. Next, the user should choose "Modify Selected Nodes..." from the "Functions" menu. This brings up the "Modify Settings" dialog box. The first entry in this dialog box is "Self Strength." The user should enter 0.5 in the appropriate box. The third entry in the dialog box is "Starting Activation Value." The user should enter 0 in the appropriate box here. At this point the initial preparations are complete. Now a description of the cyclical processing of the example text will be presented.

Processing in the contruction-integration model is cyclical. The current state of our example, though, is that all nodes are active. To model the processing of this text in a cyclical fashion the user needs to activate and deactivate nodes (and any corresponding links) in turn. This is accomplished by using the some of the commands found under the "Functions" menu. To model the first cycle of the example text, first the user must deactivate the propositions that are from cycle 2. This corresponds to propositions P4 through P7 plus the knowledge elaboration proposition A3. Use one of the methods available and select these propositions (e.g., use the pointer tool to draw a box around most or all of the propositions; use the lasso to draw a line around all the propositions; or shift-click each proposition in turn). After all the propositions have been selected, choose "Deactivate Selected Nodes" from the "Functions" menu. This will cause the selected nodes to become grayed out, both in the graphical stucture as well as the nodes' corresponding lines in the node list on the left of the screen. These nodes are now inactive. Deactivated nodes DO NOT take part in any model calculations. It is as if these nodes didn't exist for the time being.

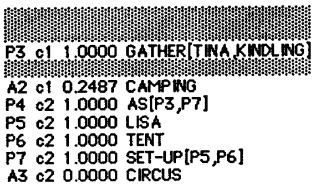
Now the nodes from cycle 1 can be integrated. To do this, choose "Iterate to Criterion" from the "Model" menu. Depending on your model of Macintosh, this may take anywhere from just a moment to a few seconds. Shortly the user should hear a tone, followed by a dialog box saying "Move results to LTM?" with a Yes and a No button. Initially it is safest to answer No to this question. A Yes response causes the LTM calculations to be performed and entered into the LTM matrix as described in many of the publications concerning applications of the model. The modeler should be familiar with the specifics of the LTM calculation before making use of this feature. Answering Yes to the dialog's question, and thus moving the results of the integration into LTM, is a step that cannot always be undone easily. In fact, to start calculating LTM values anew, it is necessary to go back to the beginning of the text in order to start over! It is best not to move the results to LTM until one's path is clear. Thus, for now answer No to this question, and then the screen should look like the adjacent figure.

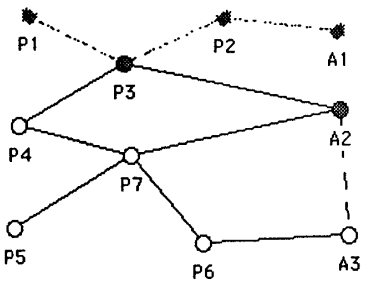




Note that the integration process took 5 iterations (using a stopping criterion of .001), and the most highly activated node is P3, or the Gather[Tina,Kindling] proposition. To perform the integration over again, simply choose "Reset Activation Vector" from the "Model" menu, which resets all the nodes' activation values back to their starting positions. To step through the integration process one iteration at a time, choose "Stepwise Iteration" from the "Model" menu. This stepwise iteration is useful for tracking the change in nodes' activation values over time. Other than this it is not particularly useful. It does not audibly point out to the modeler that the stopping criterion has been reached, nor does it allow the user to move the results of the integration process to LTM. Before moving on to the integration of the cycle two propositions of the example text, the modeler should iterate to criterion (resetting the starting activation values if need be), and this time when prompted to save results to LTM choose Yes. The user can see after doing this that the activation values are the same as they were before; the only thing that's different is that the nodes are grayed out now. This is a signal that these nodes have been entered into LTM (and thus care should be taken not to iterate to criterion again and answer Yes to "Move results to LTM?", resulting in incorrect LTM calculations being made).

Before integrating the propositions of cycle two, it is necessary to decide which proposition(s) to "carry over" to the next cycle. The number of propositions carried over is a free parameter of the Kintsch model. This number typically is small, with 1 or 2 propositions being carried over sufficient for most purposes. In the example, a buffer size of 1 is used. Thus, only proposition P3 is carried over. All other first-cycle propositions must be made inactive, except for proposition A2, which remains active because it is a knowledge elaboration for the second cycle as well as the first. For all other cycle 1 propositions (i.e., P1, P2, and A1), select them by a method described previously, and then choose "Deactivate Selected Nodes" from the "Functions" menu. Next, select all the currently inactive cycle two propositions (i.e., P4, P5, P6, P7, and A3) and make them active. The screen should now appear like the adjacent figure.





Now the nodes from cycle 2 can be integrated. Once again choose "Iterate to Criterion" from the "Model" menu. Given that this is the end of the text, it is appropriate to answer Yes to the "Move results to LTM?" question, unless, of course, the modeler wishes to go back to cycle one and carry over an additional proposition to see the effect this has on the final activation values. For now, the assumption is that the results of this cycle's integration have been moved to LTM. One result of this particular modeling endeavor can be noted by examining the final activation values for two of the knowledge elaborations. Proposition A2 has a reasonably high final activation value of 0.3006, whereas the final activation value of proposition A3 is 0.0. The model correctly predicts that the context-inappropriate association "Circus" will be deactivated as the result of a reader integrating this sentence, whereas the context-appropriate association "Camping" will be highly activated after integration .

References

Bovair, S.& Kieras, D.E. (1985). A guide to propositional analysis for research on technical prose. In B.K. Britton & J.B. Black (Eds.), Understanding expository text. Hillsdale, N.J.: Erlbaum.

Kintsch, W. (1992a). A cognitive architecture for comprehension. In H. L. Pick, P. van den Broek, & D. C. Knill (Eds.), The study of cognition: Conceptual and methodological issues (pp. 143-164). Washington, D.C.: American Psychological Association.

Kintsch, W. (1992b). How readers construct situation models for stories: The role of syntactic cues and causal inferences. In A.F. Healy, S.M. Kosslyn, & R.M. Shiffrin (Eds.), From learning processes to cognitive processes: Essays in honor of William K. Estes (pp. 261-278). Hillsdale, N.J.: Erlbaum.

Kintsch, W. (1991). A theory of discourse comprehension: Implications for a tutor for word algebra problems. In M. Carretero, M. Pope, R. Simmons, & J.I. Pozo (Eds.), Learning and instruction: European research in an international context (Vol. 3, pp. 235-253). Pergamon Press.

Kintsch, W. (1988). The use of knowledge in discourse processing: A construction-integration model. Psychological Review, 95, 163-182.

Kintsch, W., & van Dijk, T.A. (1978). Toward a model of text comprehension and production. Psychological Review, 85, 363-394.

Kintsch, W., & Welsch, D.M. (1991). The construction-integration model: A framework for studying memory for text. In W.E. Hockley & S. Lewandowsky (Eds.), Relating theory and data: Essays on human memory (pp. 367-385). Hillsdale, N.J.: Erlbaum.

Kintsch, W., Welsch, D.M., Schmalhofer, F., & Zimny, S.T. (1990). Sentence memory: A theoretical analysis. Journal of Memory and Language, 29, 133-159.

Mannes, S.M., & Kintsch, W. (1991). Planning routine computing tasks: Understanding what to do. Cognitive Science, 15, 305-342.

Miller, J.R., & Kintsch, W. (1980). Readability and recall of short prose passages: A theoretical analysis. Journal of Experimental Psychology: Human Learning and Memory, 6, 335-353.

Otero, J., & Kintsch, W. (1992). Failures to detect contradictions in text: What readers believe vs. what they read. Psychological Science, 3, 229-235.

van Dijk, T.A., & Kintsch, W. (1983). Strategies of discourse comprehension. New York: Academic Press.

Technical Report 92-14

University of Colorado at Boulder

Institute of Cognitive Science

Muenzinger Psychology Building
Campus Box 344
Boulder, Colorado 80309-0344
(303) 492-5063
FAX: (303) 492-7177
email: cogsci@psych.colorado.edu
URL: http://psychwww.colorado.edu/ics/