Chapter 4

The Reflection Assistant Model

4.1 Introduction

This research has been concerned with the creation of a reflective tool that explores new instructional designs for metacognition instruction in problem solving environments. It puts forth a model that contemplates specific metacognitive skills to be trained, proposes novel external representations of these skills, and defines diagnostic rules to assess students’ knowledge monitoring ability within ILEs. The model intends to enhance problem solving environments, reifying the metacognitive skills involved in such activities. This chapter defines the model proposed in this research, which is called the Reflection Assistant Model (or simply the RA).

Firstly, the chapter introduces the objectives of the RA. Then, it discusses the theoretical basis and the metacognitive skills modelled, and how the RA matches the domain of problem solving. The educational principles held by the RA Model are presented as well.

Next, a detailed description of the architecture and main components of the RA is offered, including the mechanics of the knowledge monitoring inference engine proposed and how the yielded results are applied to scaffolding the learning experience.

The chapter ends with a list of research questions and hypotheses put forward in this research and how we intend to investigate them.

4.2 The Objectives of the RA

The Reflection Assistant is a computational framework for metacognition instruction in problem solving ILEs. It aims at structuring the metacognitive training by defining: (i) specific metacognitive skills to be targeted, (ii) appropriate timings for this training, (iii) suitable approaches to promote the development of these skills, and (iv) mechanisms for assessing students’ metacognitive changes.

Hence, the RA intends to create appropriate conditions for students to think about themselves as problem solvers and reflect on their ongoing learning experience. This is achieved through reflective activities on those metacognitive skills considered important to attain success in problem solving.
While the RA Model is dedicated to the metacognitive aspects of learning, it has been designed to be used in conjunction with problem-solving learning environments. So, it should not be used in isolation; instead it has to be coupled to some ILE, acting as an assistant to the learning process as shown in Figure 4.1. The goals of this new *integrated* environment are:

1. Elicit a connection in the student’s mind between her metacognitive skills and her domain-level actions, as well as the products or results she generates.

2. Emphasize the importance of having an accurate judgment of the understanding of the problems to be solved as a means to improve attention and to allocate cognitive resources appropriately.

3. Demonstrate to the student means to assess, reflect on, and improve her problem solving process.

![Figure 4.1: Integration of the RA Model with problem solving learning environments. From the user’s perspective it is an integrated environment. However, the Reflection Assistant is still detached from the problem solving ILE.](image)

The Reflection Assistant Model addresses mainly the following difficulties students normally exhibit in learning situations (Hartman, 2001a): a) determining the difficulty of a task; b) monitoring their problem comprehension effectively; c) planning ahead (the RA focuses on planning metacognitive strategies); and, d) monitoring the success of their performance and their metacognitive skills employed. The RA does not focus on some of the difficulties Hartman (2001a) describes, such as: planning ahead the time they need to perform a learning task; determining when they have studied enough to master the material to be learned; using all the relevant information;
using a systematic step-by-step approach; and selecting and employing appropriate representations (see the complete list of difficulties in section 2.5.1).

### 4.3 Theoretical Basis

We have adopted Tobias & Everson’s (2002) hierarchical model of metacognition (presented earlier in Section 2.3.3) as the theoretical foundation for the RA. Their model differs from other theoretical models because it supports the view that metacognition skills could be developed in an incremental way; this makes it more appropriate for training purposes. In their pyramidal model, *knowledge monitoring* is viewed as the most basic metacognitive skill, supporting the development of other metacognitive skills shown as upper layers of the model, such as evaluation of learning, selection of strategies, and planning.

Following Tobias & Everson’s formulation of an interdependent and hierarchic relation between metacognitive skills, the RA Model proposes an incremental focus on metacognitive skills in the same order they propose. Therefore, it is primarily directed to learners’ acquisition and improvement of knowledge monitoring skill. Supported by this, it focuses on evaluation of the learning process; then on top of those two, it works on the selection of metacognitive strategies. Figure 4.2 shows our adapted theoretical model and the desired effects of training the selected metacognitive skills in a computer-based learning environment. In the next section we elaborate on the skills trained within the RA Model.

Tobias & Everson have also developed an empirically validated instrument for measuring knowledge monitoring accuracy (from now on, we call it KMA). This instrument was adapted and augmented for the purposes of our computational model and is the basis for the continuous and automatic assessment of knowledge monitoring.

![Figure 4.2: Theoretical view of metacognition in the RA Model. Adapted hierarchical model from Tobias & Everson with the metacognitive abilities that are emphasized in our research.](image-url)
4.3.1 Metacognitive skills modelled

Our computational framework focuses on the following metacognitive skills: knowledge monitoring, evaluation of the learning experience, and selection of metacognitive strategies.

The RA does not include “planning”, located at the top of Tobias & Everson’s model, because we preferred to limit the scope of the framework. We can hope, however, that the training on those three fundamental skills would trigger the autonomous development of planning and other metacognitive skills.

Modelling knowledge monitoring

Scaffolding knowledge monitoring is the prime target of the RA Model. Knowledge monitoring is the ability to assess one’s knowledge or, by extension, one’s understanding.

The RA is built on the assumption that promoting awareness of one’s level of knowledge monitoring accuracy is the first step to foster this skill. Improvements in knowledge monitoring ability in turn triggers selective attention and facilitates better allocation of cognitive resources. That is, having identified what one does not know but should know, help to shift one’s attention to it. In a problem solving scenario, knowledge monitoring targets the understanding of the givens and the goals of the problem. An effective assessment of one’s understanding of these elements may lead to further readings of the problem description for a deeper understanding of what exactly is being asked and what information is provided in the problem.

Modelling evaluation of learning

As seen in Chapter 2, research on problem solving supports the idea that when the student steps back and reflects on how she actually solved a problem, she is putting attention directly on her own thought process (Brown, 1985). Thus, the activities proposed in our model aim to develop awareness of how one behaves during problem solving, which resources one uses, how long one spends on each task, and which decisions one makes in the process of solving a problem.

As a result it is expected that the student will make better plans in future learning interactions, based on conscious knowledge of what works best for her and what could be improved. For example, the student could approach future problems knowing that her habit of taking her time reading the problem is a good thing, while she needs to spend more time checking the solution once it is shown to her.

Modelling selection of metacognitive strategies

Finally at the top level of our model comes the ability to select good strategies to be applied during the learning process. The focus here are the metacognitive strategies, i.e. general heuristics or strategies that are loosely connected to the domain and the task. The RA Model focuses on developing students’ awareness of three kinds of metacognitive strategies: strategies for monitoring understanding, strategies for monitoring the problem solving process and controlling errors, and strategies for revising.

4.3.2 Domain of application: ILEs for problem solving

The RA is designed to be used in conjunction with problem solving learning environments. Thus, it needs to integrate quite tightly with this kind of environment in order to adapt the learning process, create new meaningful activities and learning targets.
As presented previously in Section 2.8 the problem solving activity can be divided into three conceptual stages: (a) preparation to solve the problem or familiarization stage, (b) production stage, and (c) evaluation or judgement stage. Therefore, the RA Model is organized around these stages, matching specific metacognition instruction to the characteristics of each of these stages. Figure 4.3 shows a high level diagram of the association of the RA Model with the problem solving stages.

Figure 4.3: Problem solving conceptual stages and the RA Model. Two new stages are proposed: pre-task reflection and post-task reflection.

At the top of the diagram is a timeline representing a typical problem solving episode broken down into conceptual stages. The layer in the middle of the diagram shows the cognitive skills that are brought to bear on the process as time goes by. The layer at the bottom represents the metacognitive skills which are active along the way.

Considering these conceptual stages, we have set specific moments where each of the selected metacognitive skills shall be trained. As such, knowledge monitoring and selection of metacognitive strategies (skills defined in Section 4.3.1) are mainly explored in the familiarization stage, when the learner should naturally spend some time understanding the nature of the problem, recognizing the goals, and identifying the givens of the problem.

We believe that cognitive load is higher at the production stage. Moreover, as observed in previous empirical studies (Rutz et al., 1999, for example) knowing when to interrupt students during problem solving, and having a good match between the content of the reflection and the current task are key to make them reflect. Unfortunately that kind of match is impossible to do given that we will not have access to the thoughts of the student.

Therefore, the design of the RA Model does not include any major interference during the production stage. Instead, two new conceptual stages in problem solving were created, which are called pre-task reflection and post-task reflection.
These stages can be seen as extensions of, respectively, the familiarization stage and the evaluation stage (look again at the timeline depicted in the diagram). At these new stages the cognitive load is lower, because the student is not engaged in actually solving the problem, but still has the problem fresh on her mind (in the case of pre-task reflection, this will be a problem solved earlier).

Thus, the RA Model uses this “cognitive space” to promote reflection on knowledge monitoring and evaluation of learning, in this case of the problem solving experience (see definition of evaluation of learning in Section 4.3.1).

As such, the metacognitive skills come to the foreground, becoming the object of reflection.

**Pre-task reflection stage**
Self-reflection is normally regarded as an after task activity. In the RA self-reflection also happens before the learning task. It covers the metacognitive aspects necessary to start the new problem; it provides suitable conditions for the student to become aware of useful general strategies, resources available, and also the degree of attention necessary to succeed in solving the problem.

Collins and Brown (1988) affirm that sometimes self-reflection does not occur naturally, in part because students are not really motivated to perform reflection after solving the problem and in part because the current problem solving medium (paper and pencil) does not really lend itself to this activity. By placing self-reflection before the learning activity, we expect to reduce problems of low motivation.

**Post-task reflection stage**
Self-reflection after problem solving has a different goal. It involves mainly the most recently problem attempted, but it is also linked to the student’s reflections before the beginning of the problem. In other words, the post-task reflection stage creates a space where the student thinks about her actions during the past problem solving activity, comparing them with her reflections expressed just before the beginning of that problem. Using the task in hand the student abstracts her thoughts to a metacognitive level, thinking about herself as a problem solver and about her general learning processes. The goal is to trigger self-questioning about the learning experience such as: Did I use all the problem solving resources available? How well did I manage my time? Did I spend enough time reading the problem and making a suitable plan of action? What could I have done differently?

**Exemplifying the matching of the RA Model with a problem solving activity**
We regard maths problem solving environments as the main candidate for adopting the RA Model. Accurate knowledge monitoring, attentive evaluation of learning and a good selection of metacognitive strategies play important roles in helping students to become good maths problem solvers.

Understanding mathematical text problems is a complex and knowledge-intensive inferential and highly constructive process that requires skillful interaction of more than one kind of knowledge, including linguistic, situational, as well as mathematical knowledge (Reusser, 1993). Knowledge monitoring provides awareness of problem comprehension and it is the first step towards the search for a deeper understanding.

Polya (1945) states that progress toward a maths problem goal is incremental and each move should be informed by a repertoire of strategies and heuristics. Furthermore, Schoenfeld (1992) says that heuristics should be taught in the context of a metacognitive strategy that supports the
learner in selecting the right one to solve a problem. We believe that being aware of the heuristics one possesses and consciously applying them is key to improve learning. Thus, the selection of metacognitive strategies in the RA follows Polya and Schoenfeld’s principles to improve heuristic search.

Finally, research on maths problem solving affirms that it is important to reflect on the steps taken in order to evaluate the use of cognitive and instructional resources. Reflection on adequate allocation of time, effort, usage of prior knowledge and heuristics learnt are important objects of evaluation of the problem solving experience. Promoting awareness on the whole experience makes students realize that solving a problem is not a linear activity, but a complex knowledge building endeavour.

4.4 Design of the Reflection Assistant Model

One possible general view of the RA Model is through the classification schema we have introduced in Chapter 3. The dimensions are: timing, that refers to the moment chosen for promoting training on metacognitive skills; target level, the refers to the degree of connection of the metacognitive skills trained with the domain; and instructional approach, which refers to the representation mechanisms chosen to provide the training. The possible values for this classification were presented before and are summarised in Figure 3.1. According to this classification, the RA Model can be categorize as shown in Figure 4.4 below.

![Figure 4.4: The RA in our classification of metacognitive training.](image)

The RA proposes reflective activities both before and after the problem solving task, and during the familiarization stage of problem solving, but not during the production and evaluation stages. The target level is on domain-related skills (skills that are relevant to problem solving in general) and task-specific skills (skills that are relevant to solve the specific problems presented). Three instructional approaches were adopted: graphical reification combined with metacognitive scaffolding, and self-assessment.

The RA combines the elements in these dimensions in different ways, depending on the situation. For example, the reflective activity related to knowledge monitoring is proposed before the problem solving task and uses graphical reification of student’s assessment of her understanding
of previous problems performed, together with metacognitive scaffolding messages.

4.4.1 The Architecture of the RA Model

The RA is kept as general as possible so that it can be adapted according to the specific domain and ILE to which it will be incorporated. Figure 4.5 presents the general architecture of the Reflection Assistant Model. Similarly to Figure 4.1 shown before, this diagram depicts the communication of the RA Model with the problem solving learning environment and the interaction of the user with both environments.

![Figure 4.5: Architecture of the Reflection Assistant Model. Main components, information bases, and interaction with the problem solving learning environment.](image)

The RA is divided into two main modules, which are called: *pre-task reflection and familiarization assistant* and *post-task reflection assistant*. Students interact with the activities proposed within the modules, which aim to provide structured support for metacognition development.

The *pre-task reflection and familiarization assistant* is used in the newly created pre-task reflection stage and also in the familiarization stage of problem solving. The aim of this module is to prepare the student for the problem solving activity, promoting reflection on knowledge monitoring, assessment of the difficulty and understanding of the problem to be attempted, and awareness of useful metacognitive strategies.

The activities labelled as “Comparison of knowledge monitoring and performance” and “Analysis of knowledge monitoring state (KMA/KMB)” are not related to a specific problem; rather they refer to the overall experience of the student with the problem solving environment, and uses information from previous interactions with the RA Model and previous problems performed. The
activities labelled as “Assessment of problem comprehension and difficulty” and “Selection of metacognitive strategies” are related to the current problem the student is about to attempt. They take place in the familiarization stage of problem solving, just before the student starts to actually solving the problem.

The post-task reflection assistant presents activities related to the evaluation of problem solving. As the name suggests, it takes place in the post-task reflection stage, just after the student finishes a problem. It involves two activities labelled “Evaluation of problem solving experience” and “Self-assessment of strategies usage”. After concluding a problem the student interacts with this assistant to analyse her performance, her use of problem solving resources, time management, and use of strategies.

Besides the modules, the RA includes an inference engine to assess students’ level of metacognition. The engine is an algorithm that uses formulas and rules to infer knowledge monitoring ability. The engine uses information from students’ interaction with the problem solving environment and from the RA Model and generates two measures: knowledge monitoring accuracy (KMA) and knowledge monitoring bias (KMB). These concepts and their measures will be presented in detail in the next section.

Finally, the RA incorporates a series of data repositories. The “library of metacognitive feedback” and the “library of metacognitive strategies” contain general knowledge about metacognition, and the “student KMA/KMB model”, the “student’s metacognitive strategies”, and the “history of student’s reflective actions” are repositories of information about students’ demonstrated or inferred metacognition.

As seen in the architecture diagram, the communication between the RA Model and the ILE is through two data repositories in the ILE side: “student’s problem solving performance” and “history of student’s problem solving actions”. So, even detached from the problem solving environment, the RA needs information about students’ actions that comes from the ILE. Thus, in order to use the RA Model it is essential that the ILE implements these repositories. In the section about the repositories we detail all pieces of information that have to be provided for both the RA and ILE.

From the user’s perspective, the RA Model may appear either separated from the problem solving environment (e.g. in specific and separated windows) or it can be embedded in the ILE. Independently of the choice, the interaction takes place in the following sequence: (1) the student starts by performing the first two activities proposed in the pre-task and familiarization assistant, then the ILE presents a new problem and she proceeds to the other two activities of the the pre-task and familiarization assistant; (2) she solves the problem with the aid of the problem solving tools provided by the ILE; and (3) after finishing the problem, she performs the activities proposed by the post-task reflection assistant.

4.4.2 Metacognitive inference engine

The RA Model incorporates an assessment mechanism of students’ metacognition and uses it to maintain a student metacognitive model. It is a simple inference engine, since it infers only one metacognitive skill: knowledge monitoring.

The student’s knowledge monitoring ability is inferred from the ILE (using the student’s per-
formance on problem solving) and from the RA (using the student’s own prediction of her understanding of problems and ability to solve them).

The inference mechanism is applied every time the student attempts a new problem and the student model is updated as a result. The information comprised in the student model is then used in the pre-task reflection activities provided by the RA.

Two aspects of knowledge monitoring ability are inferred: knowledge monitoring accuracy (KMA) and knowledge monitoring bias (KMB).

The Knowledge Monitoring Accuracy (KMA) refers to how skillful a student is at predicting how she will perform on a learning task; it reflects her awareness of the knowledge she possesses.

The Knowledge Monitoring Bias (KMB) provides a statistical measure of any tendency or bias in the learner knowledge monitoring ability.

The assessment mechanism of these measures and rules for classifying students’ KMA and KMB scores are shown next.

4.4.3 Definition of the Knowledge Monitoring Accuracy (KMA) measure

We have developed a formula for measuring knowledge monitoring accuracy (KMA) based on Tobias & Everson’s assessment instrument (presented in Section 2.4.2). Their original instrument evaluates the student’s knowledge monitoring ability by first asking her whether she is able to solve a problem and later asking her to solve that problem. The KMA results from the match between these two pieces of information.

By collecting a significant number of elementary assessments for the same student, their instrument produces a statistical profile of the student’s awareness of her own knowledge. The scale of possible scores for KMA varies between -1 and 1, where -1 signifies very low accuracy of knowledge monitoring, and 1 means complete accuracy. Values around 0 signify a random assessment; in other words, that the student failed in her assessment as often as she succeeded. They present the following formula for scoring the final KMA:

\[
\frac{((a + d) - (b + c))}{(a + b + c + d)}
\]

where \(a\) is the number of times the student predicted success and succeeded, \(b\) is the number of times the student predicted failure and succeeded, \(c\) is the number of times the student predicted success and failed, and \(d\) is the number of times the student predicted failure and failed (for further explanation, please refer back to Section 2.4.2).

To build our KMA measure we reinterpreted this formula as the mean of individual scores evaluated for each prediction/performance cycle. Each such score, which we call \(kma\), represents the correctness of the prediction on that particular cycle.

As there are two possible values for the evaluation and two for the performance, \(kma\) has to be defined for a total of four cases. Tobias & Everson’s formula is obtained by giving \(kma\) the following values:

<table>
<thead>
<tr>
<th>Performance</th>
<th>Estimation</th>
<th>Evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>correct</td>
<td>will solve</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>will not solve</td>
<td>-1</td>
</tr>
<tr>
<td>not correct</td>
<td>-1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
We see that \( kma \) is 1 when the assessment is correct (i.e. predicted that was not able to solve the problem and indeed did not solve it correctly, or predicted that was able to solve the problem and solved it correctly) and -1 when the assessment is incorrect (i.e. predicted that was able to solve the problem, but did not solve it correctly, or predicted that was not able to solve the problem, but solved it correctly).

If we average \( kma \) over a series of problems, we end up with the KMA score as proposed by Tobias & Everson.

Seeing this formula as an average, we notice that in its original form, the whole KMA scheme is a binary one. Prediction and performance are binary and the variable \( kma \) also ends up taking only two distinct values. We wanted to have a bit more flexibility and be able to differentiate between students who did not solve the problem correctly at all from those who presented a partially correct solution path. In order to do that, the minimum change we could do was to allow both prediction and performance to take a third value representing some intermediary state.

**Defining our new measure of KMA**

In the dimension of prediction, we added the possibility for the student to predict that she would partially solve the problem or that she partially understood it. In the dimension of performance, we now treat partially correct answers as a meaningful case. The 2x2 matrix needs to become a 3x3 matrix to accommodate all possible combinations as shown in Table 4.1.

We then had to decide which values \( kma \) would take for each of the 9 cases so defined. The four corners of the new matrix are equivalent to the original four cases of Tobias & Everson’s table. It was thus natural to use the same values for them.

The point in the middle is similar to the top left and bottom right corners in that the prediction and the performance are matched. It therefore makes sense to assign 1 to \( kma \) in this case as well.

All the other cases represent mismatches between what was predicted and what happened. However these mismatches are not as large as the ones corresponding to the bottom left and top right corners of the matrix. For instance, when the student predicts that she will solve the problem partially and solves it fully, this is a knowledge monitoring error. However it is not a bad an error as if she had predicted that she would not solve the problem at all. It therefore makes sense to associate with such mistakes a \( kma \) value larger than -1, which represents the full mistake.

At first glance, a good candidate would be 0 which is just half way between -1 (being completely wrong) and 1 (being completely right). However, doing so would mean that students that always make such small mistakes in their assessments remain with a null value of KMA. Only one correct assessment would then push their KMA to a positive value. It seemed more appropriate to have two such small mistakes cancel out one accurate assessment. For that to happen, we decided to assign -0.5 to all those 4 cases.

Thus, one requisite for the ILE is that it analyzes and breaks down the student’s performance into the three possible values: correct, partially correct, or incorrect. Our metacognitive inference engine will need such a categorization to function. Similarly, the RA has to gather estimates of performance with the three possible values: able to solve completely, able to solve partially, not able to solve the problem.

The mean of \( kma \) scores over all problems solved so far yields the **current KMA state** of the student. The more the student interacts with the ILE and RA, the more reliable becomes the RA’s
Chapter 4. The Reflection Assistant Model

Prediction $\rightarrow$ Performance

<table>
<thead>
<tr>
<th></th>
<th>estimated will</th>
<th>estimated will</th>
<th>estimated will</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>not solve</td>
<td>solve partially</td>
<td>solve</td>
</tr>
<tr>
<td>demonstrated not correct</td>
<td>1</td>
<td>-0.5</td>
<td>-1</td>
</tr>
<tr>
<td>demonstrated partially correct</td>
<td>-0.5</td>
<td>1</td>
<td>-0.5</td>
</tr>
<tr>
<td>demonstrated correct</td>
<td>-1</td>
<td>-0.5</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 4.1: Matrix of values for Knowledge Monitoring Accuracy (KMA).

Students’ Classification of KMA

The score inferred for the KMA is shown to students in the reflective activities. For this purpose the numeric values are converted into qualitative ones. The classification summarises scores by mapping them in three categories: low KMA, average KMA, and high KMA. Table 4.2 shows the exact ranges of this mapping. So, if a student has a KMA of 0.8, she will be assigned a high KMA profile, which means that most of the time she demonstrates accurate estimations of her knowledge.

Notice that we have voluntarily made the high range smaller than the lower ranges, because we decided that we would acknowledge as high KMA only those students with a very consistent accurate behaviour.

<table>
<thead>
<tr>
<th>KMA Value</th>
<th>Classification</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-1, -0.25)</td>
<td>Low KMA</td>
<td>Learner doesn’t estimate correctly her knowledge in the majority of situations</td>
</tr>
<tr>
<td>[-0.25, 0.5)</td>
<td>Average KMA</td>
<td>Learner sometimes estimates correctly, but makes frequent slightly wrong or completely wrong estimations</td>
</tr>
<tr>
<td>[0.5, 1]</td>
<td>High KMA</td>
<td>Learner most of the time makes correct estimation of her knowledge</td>
</tr>
</tbody>
</table>

Table 4.2: KMA classification. Students’ categories according to their KMA scores.

4.4.4 Definition of the Knowledge Monitoring Bias (KMB) measure

The Knowledge Monitoring Bias (KMB) measure was created since the KMA does not provide a detailed account about the type of inaccuracies the student may show. For example, imagine a student that was assigned a low KMA profile, because she is constantly predicting that she will solve the problems correctly, but her problem solutions are invariably wrong. This case is different from the one of another student that tends to estimate that she will not solve the problems
completely correct, but then most of the time she reaches a correct solution. So, in order to describe
the pattern or bias in the learner’s knowledge monitoring ability we developed the KMB measure.

The KMB takes into account the way student deviate from an accurate assessment of her
knowledge monitoring. If there is no deviation, we say that the student is accurate in her assess-
ment of her knowledge or realistic about it. Otherwise, three cases are possible: (i) the student
often predicts she will solve the problems but she does not succeed, demonstrating through this an
optimistic assessment of her knowledge; (ii) the student often predicts she will not solve the prob-
lems, but then she succeeds in solving them, demonstrating through this a pessimistic assess-
ment of her knowledge; and (iii) she is sometimes optimistic in her assessment of her knowledge as she
is pessimistic, in a random pattern.

The KMB scores are dependent of the KMA demonstrated. So, the possible range of val-
dues for the KMB is also from -1 to 1. Similarly to the KMA, a 3x3 matrix was also created for
providing the possible scores for each individual kmb (see Table 4.3). The possible kmb scores
have a different meaning from the kma. Accurate estimations receive the score 0, meaning a null
bias; 1 is assigned to the highest optimistic estimation; likewise -1 is assigned to the highest pes-
simistic estimation; -0.5 is assigned to a minor pessimistic estimation and 0.5 to a minor optimistic
estimation.

<table>
<thead>
<tr>
<th>Prediction →</th>
<th>estimated will solve</th>
<th>estimated will solve partially</th>
<th>estimated will solve</th>
</tr>
</thead>
<tbody>
<tr>
<td>demonstrated not correct</td>
<td>0</td>
<td>0.5</td>
<td>1</td>
</tr>
<tr>
<td>demonstrated partially correct</td>
<td>-0.5</td>
<td>0</td>
<td>0.5</td>
</tr>
<tr>
<td>demonstrated correct</td>
<td>-1</td>
<td>-0.5</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 4.3: Matrix of values for Knowledge Monitoring Bias (KMB).

Students’ Classification of KMB
A classification of student’s current KMB state in respect to her predictions deviations is made
based on the mean of kmb scores over all problems solved so far.

As the score inferred for the KMB is used in the reflective activities of the RA Model, a
classification that converts the numeric values into qualitative ones was defined to make it easier
to the student to understand her bias. The classification summarises scores by mapping them in
four categories: realistic, optimistic, pessimistic, and random. Table 4.4 shows the exact ranges of
this mapping.

So, if the learner was assigned a value that represents a “high KMA”, she is then classified
as realistic and it means that she has no bias in her judgement of her knowledge. However, if
she has obtained an “average KMA” or a “low KMA”, it means that she has some kind of bias in
her judgement and a KMB value is calculated. For these cases, the KMB value is matched to the
corresponding interval. For example, if the student has a low KMA (meaning a low accuracy of her knowledge monitoring) and has a KMB equals to - 0.7, then it is said that the student tends to have a pessimistic assessment of her knowledge, often saying that she is not able to solve the problems, but actually succeeding in solving them or partially solving them.

<table>
<thead>
<tr>
<th>KMB Value</th>
<th>Classification</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>High KMA</td>
<td>Realistic</td>
<td>Learner makes an accurate estimation of her knowledge, having a High KMA</td>
</tr>
<tr>
<td>[0.25, 1]</td>
<td>Optimistic</td>
<td>Learner tends to estimate she can solve the problems, but she does not succeed in the majority of situations</td>
</tr>
<tr>
<td>[-1, -0.25]</td>
<td>Pessimistic</td>
<td>Learner tends to estimate she can not solve the problems, but then she succeeds</td>
</tr>
<tr>
<td>(-0.25, 0.25)</td>
<td>Random</td>
<td>Learner estimates of her knowledge are as often optimistic as pessimistic</td>
</tr>
</tbody>
</table>

Table 4.4: KMB classification. Students’ categories according to their KMB.

By exhibiting to the student her KMA and KMB levels the RA intends to promote awareness and development of knowledge monitoring skill.

4.4.5 Data repositories and activities

This section presents the other components of the RA: data repositories and activities within modules. The description of the activities was organized into the following rubrics: what: shows the objectives of the activity; how: presents the approach devised with the interface mock-ups suggested; input/output: presents the input data used and the output data generated by the activity.

The reflective activities use graphical representations associated to textual explanations to convey information about metacognitive skills and the learning processes. The diagrams and graphs proposed are believed to transform the reflective activities into easier, perceptual ones. Considerable research has taken place on the different mechanisms through which graphical representations assist users in drawing inferences, solving problems, comprehending computer programs, etc. (Larkin and Simon, 1987; Tufte, 1990; Cox, 1999). In general it has been accepted that a graphical method of representing information and solution improves performance, individual understanding, and solution quality (Larkin and Simon, 1987). Hence, we use graphical representations of the learners’ metacognitive abilities and learning processes to help them see patterns, express abstractions in concrete form, and discover new relationships.

Data repositories

A list of all pieces of information that is stored in the RA repositories are presented in the Table 4.5. The data is presented in a general degree of detail.

Table 4.6 enumerates the repositories that have to be created in the ILE in order to provide information about students’ actions and performance to the RA.
The RA reflective activities

**Activity 1: Comparison of knowledge monitoring and performance**

**What:** This activity aims to trigger reflection on student’s trend and progress of her knowledge monitoring ability. It focuses on past problems, showing to the student her previous performance and comparing them to her judgements of her knowledge and understanding of those problems.

**How:** It replays to the student her previous demonstrated performance (summarised into the following categories: low, average, and high) for all past problems solved together with the her prediction of solving those problems (also summarised into: low, average, and high). It uses a graphical reification such that the student can compare the two pieces of information. Figure 4.6 shows the mock-up made for this graphical reification; it depicts bar-type graphs showing the assessment of problem understanding next to the performance for all past problems. Beside the graphical display, textual explanations are necessary to provoke appropriate interpretations of

<table>
<thead>
<tr>
<th>Table 4.5: Data repositories of the RA Model.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Student KMA/KMB Model</strong></td>
</tr>
<tr>
<td>For each problem:</td>
</tr>
<tr>
<td>Assessment of the difficulty to perform it</td>
</tr>
<tr>
<td>Prediction of solving it correctly</td>
</tr>
<tr>
<td>KMA score calculated after problem performed</td>
</tr>
<tr>
<td>KMB score calculated after problem performed</td>
</tr>
<tr>
<td>Student’s reflections on her metacognitive progress and tendency</td>
</tr>
<tr>
<td><strong>Student’s Metacognitive Strategies</strong></td>
</tr>
<tr>
<td>For each problem:</td>
</tr>
<tr>
<td>List of strategies selected (they can be created or edited from the library of metacognitive strategies)</td>
</tr>
<tr>
<td>Student’s reflections on the use of the strategies selected</td>
</tr>
<tr>
<td><strong>Library of Metacognitive Feedback</strong></td>
</tr>
<tr>
<td>Texts with instructional scaffolding messages related to each metacognitive value for the KMA and KMB</td>
</tr>
<tr>
<td>Texts with instructional scaffolding messages related to strategies selection</td>
</tr>
<tr>
<td>Texts with instructional scaffolding messages related to evaluation of the learning experience</td>
</tr>
<tr>
<td><strong>Library of Metacognitive Strategies</strong></td>
</tr>
<tr>
<td>List of strategies for monitoring understanding</td>
</tr>
<tr>
<td>List of strategies for controlling errors</td>
</tr>
<tr>
<td>List of strategies for revising</td>
</tr>
<tr>
<td><strong>History of Student’s Reflective Actions</strong></td>
</tr>
<tr>
<td>For each problem, time spent on reflective activities both before and after the problem</td>
</tr>
</tbody>
</table>
**Student’s Problem Solving Actions**

For each problem, whether the solution given was correct, partially correct or not correct.

<table>
<thead>
<tr>
<th><strong>History of Student’s Problem Solving Actions</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>For each problem, the time spent on each problem solving resource (date and time of the beginning and date and time of the end)</td>
</tr>
</tbody>
</table>

**Table 4.6:** Data repositories provided by the ILE.

The graph, asking students to look for trends, to check for improvements, and to self-explain the reasons for the differences shown (if they exist).

Larkin and Simon (1987) describe the benefits of using both diagrams and text to describe a scene, a 3-dimensional object, etc. We believe that a combination of visualizations and text is a fruitful way to facilitate learning of metacognition. For example, the text would be helpful at a very early point when a student is just learning to interpret the graphical representation of her predicted and demonstrated performance and infer patterns from that.

**Figure 4.6:** Mock-up of the graphic component of the first pre-reflection activity. Graphical reification of the comparison of predicted knowledge and performance.

**Input:**

- Prediction of solving each problem correctly.
- Performance in problem solving.

**Output:**

- Student’s reflections on her metacognitive progress and tendency.
- Time spent on the reflective activity.
- Identification of the reflective activity.
Activity 2: Analysis of knowledge monitoring state (KMA / KMB)

What: This activity also focuses on knowledge monitoring ability. Differently from the previous activity, it does not show past problems performance; it refers to problem solving in general and presents the RA’s inferred assessment of student’s level of KMA and KMB. In this way, it aims to foster accuracy of knowledge monitoring by raising awareness of and initiate discussion on knowledge monitoring ability overtly.

How: The values calculated for the KMA and KMB by the inference engine are converted into degrees (KMA= high, average, low / KMB= realistic, random, optimistic, pessimistic) and a visual display shows these levels in a coloured and easy to grasp manner. We have devised graphical widgets similar in form to a car speedometer and which we call reflectometers.

Similarly to the “skillometers” of Anderson et al. (1992) that were designed to present learners with a general view of the state of their knowledge of the domain, the reflectometers present learners with a view of their knowledge monitoring ability.

Figure 4.7 shows the mock-up made for the reflectometers and other possible interface functionalities. The first reflectometer indicates the accuracy of the student estimate of her own knowledge (KMA) and the second shows any bias that exists in that estimation (KMB).

![Figure 4.7: Mock-up of graphic component of the second pre-reflection activity. Graphical reification for the analysis of knowledge monitoring state.](image)

Textual explanations and scaffolding are very important in this activity, since the student needs some help to interpret the information provided in the reflectometers and guidance to adjust her future problem solving actions to improve her knowledge monitoring scores. The scaffolding should be tailored to the values shown and also based on previous actions the student has taken (e.g. if the time spent on assessing problem comprehension is generally low, one possible instructional scaffold could be “try to spend more time reading the problem and carefully evaluating your understanding of the concepts involved, before stating your understanding of the problem”).

Input:

- mean of the KMA using all KMA scores calculated after each problem performed.
- mean of the KMB using all KMB scores calculated after each problem performed.
Output:

- Time spent on the reflective activity.
- Identification of the reflective activity.

**Activity 3: Self-assessment of problem comprehension and difficulty**

**What:** This activity is related to the self-assessment of the current problem still to be solved. It aims to make the student reflect on her understanding of the concepts and components of the problem and on her confidence to solve it correctly. It also gathers information that will be used later on the future pre-reflective activities (Activities 1 and 2).

**How:** Questions about the student’s level of comprehension of the problem and difficulty to solve it are presented. Discrete scales should be used. The number of questions depends on the domain and level of detail it will be necessary to have a good picture of the student’s understanding of the problem. Examples of questions are: *Do you recognize the type of the problem? Do you understand the goals of the problem? Have you done a similar problem before?* etc. It is important that the student provides all the information requested prior to starting to attempt to solve the problem. There is no right or wrong answer; the answers must reflect the student’s judgement of her understanding.

**Input:** This activity does not use any input data from the RA Model. It only uses the problem description that is provided by the problem solving ILE.

**Output:**

- Assessment of the difficulty to perform the current problem.
- Prediction of solving the current problem.
- Time spent on the reflective activity.
- Identification of the reflective activity.

**Activity 4: Selection of metacognitive strategies**

**What:** The goal of this activity is to make students reflect on strategies that can be useful to solve the problem at hand. It focuses on metacognitive strategies related to the three metacognitive processes that should take place during problem solving: monitoring understanding, monitoring progress and controlling errors, and revising solution paths. Thus, this activity helps students to think of relevant strategies, their purposes and appropriate moments to apply them.

**How:** Lists of pre-defined strategies are presented to the students and they are asked to select those they think may help them to solve the current problem. Besides that, students should be allowed to create their own strategies and also to edit the existing ones. Extra functionalities can be incorporated, such as ordering the strategies by importance or relevance, identification of strategies that are always used for all problems, etc.
Chapter 4. The Reflection Assistant Model

The RA does not interfere with the problem solving stage. However, one possible option is to present the list of selected and created strategies in a separate window during problem solving. This will create minimal interference and may serve as a reminder to the student or a trigger to make them apply those strategies. Another possibility is to offer the list whenever the student asks for help. The designer is free to implement the solution that appears to be more convenient to the domain and goals of the problem solving environment.

**Input:**

The activity uses as input data the information in the library of metacognitive strategies. The library contains specific strategies, organized into three groups: strategies for monitoring understanding, strategies for controlling errors, and strategies for revising. Below we suggest some strategies to be included in the library. They were taken or adapted from the literature on metacognition assessment (Schraw and Dennison, 1994; Pintrich et al., 1993).

- **Strategies for Monitoring Understanding:**
  1. Read the problem more than once.
  2. Read the problem to separate important parts or identify components.
  3. Think of a related problem you have already done and use it as a model.
  4. Before starting to solve the problem, think what you are supposed to learn from it.
  5. Read the problem and determine which parts you don’t understand well.
  6. Review the basic concepts that are not clear before attacking the problem.
  7. Set a goal to yourself and plan the steps to reach this goal.

- **Strategies for Controlling Errors:**
  1. Stop and review each step to see if you made a mistake.
  2. Reread the problem from time to time to check for forgotten important parts.
  3. Stop and change strategies if you get lost and confused and don’t seem to move anywhere.

- **Strategies for Revising:**
  1. Think about a way of checking to see if your solution is correct.
  2. Review all you did to make sure you are not forgetting anything.
  3. Reread the task description and ask yourself if your solution really meets the task goal.

**Output:**

- Strategies selected, edited or created by the student for the problem.
- Time spent on the reflective activity.
- Identification of the reflective activity.

**Activity 5: Evaluation of problem solving experience**

**What:** Similarly to the reflection-on-action proposed by Schön (1987), this activity aims to give an opportunity to the student to review her most recent experience, exploring why she acted as
she did, what happened during problem solving, etc. The focus is on helping her to reflect on the “causes of her mistakes” related to the process, her use of resources, and time management issues. In so doing she can develop a better understanding of her problem solving experience and practice.

**How:** We have developed a graphical reification of the student’s interaction with the problem solving activity. Figure 4.8 shows the mock-up for the graphical component of the activity. The graphical reification proposed was inspired in Schoenfeld’s timeline graph originally created for analysing students’ problem solving behaviours (as seen in Figure 2.5). We envisioned a timeline graph that shows the problem solving activities and the reflective activities.

Looking at the graph the student can observe when (and if) she used the main problem solving resources available, the number of times the resources were used for a given problem, the time spent on reflective activities, etc. Another possible option is to show the moments when she asked for help. A textual explanation of the information depicted by the graph is also needed to trigger reflection on the main “conflicts” that may exist on the student’s performance (e.g. the student said the problem was difficult, but then she did not use any of the resources provided by the problem solving environment to help on solving the problem, or she spent little time selecting strategies to solve the problem, etc.). Finally, it could also be beneficial to provide a comparison of the student’s performance in the last problem solved to the one of a more experienced person for the same problem, or to compare the problem to a previous similar problem done by the student. Figure 4.9 presents an initial mock-up for this last option.

**Input:**

- History of student’s reflective actions (duration of each activity).
- History of student’s problem solving actions (duration of each activity).
- Student’s problem solving performance.
Figure 4.9: Mock-up of optional graph to compare performance between similar problems. Also used in the evaluation of problem solving experience activity.

- Assessment of difficulty to perform each problem.
- Student’s prediction of solving each problem.
- Metacognitive texts - feedback.

Output:

- Time spent on the reflective activity.
- Identification of the reflective activity.

**Activity 6: Self-assessment of strategies usage**

**What:** The goal of this activity is also to reflect on the most recent problem solving experience, but this time the focus of the reflection are the metacognitive strategies the student planned to use. The activity helps the student to review the experience from this point of view and to analyse whether she applied or not a strategic behaviour during problem solving.

**How:** The list of metacognitive strategies selected or created by the student at the beginning of the problem is presented again and some reflective questions are posed concerning the use of those strategies during problem solving. The questions serve as a trigger for making the student self-explain if and when she used the strategies and whether they were useful or not during problem solving. Questions like: Did you use the strategies during problem solving? Which ones were most useful? When did you use them? What would you do differently next time? are helpful to guide student’s self-explanations.

**Input:**

- Strategies selected, edited or created by the student for the problem.
Output:

- Student’s reflections on the use of the strategies selected for the problem.
- Time spent on the reflective activity.
- Identification of the reflective activity.

4.4.6 Adding the RA to an existing ILE

The Reflection Assistant model is a generic framework whose principles should be applicable to any kind of problem solving ILE.

In order to facilitate integration with existing ILEs, we have minimized the coupling between the RA and the ILE. Information flows only one way from the ILE to the RA. As shown on Figure 4.5, the RA only needs access to two well defined data repositories: one keeps information on the student’s performance, the other on the history of the interaction between the student and the resources of the problem solving environment. If the ILE keeps track of the interaction with the student in some form, it ought to be easy to add a wrapper around the existing datastore to get it to conform to the interface needed by the RA. More work will be needed in the case where the ILE doesn’t keep any history of its interaction with the student. In such case, this datastore will have to be built from the ground up.

In terms of user interface, the mock-ups presented are only a guideline for the final graphical components. They would have to be adapted so that the phrasing of messages is relevant to the ILE. Whether they are included in existing windows or included in new additional windows will also have to be decided on an ad-hoc basis.

4.5 Research Questions and Hypothesis

The RA model is based on the notion that focusing on metacognitive skills as object of reflection triggers the development of these skills and has the potential to improve learning. Hence, it intends to make students aware of the importance of metacognition for the success of the learning endeavour.

More specifically, the RA promotes the idea that by reflecting on one’s own knowledge monitoring skills, one increases one’s own understanding of the influence of these skills in the outcomes of the problem solving experience.

Therefore, our main research question can be expressed as: Does the Reflection Assistant Model help students to improve their metacognition? And, if so, does it lead to improvements in problem solving performance?

We expect that the interaction with the reflective activities proposed by the RA encourages students to become more conscious about their learning processes and skills. Thus, the hypotheses driving this research are that reflection on the metacognitive skills between problem-solving learning activities has the potential to:

(i) enhance students’ knowledge monitoring skills and in turn this helps them to become more aware of how to allocate their cognitive resources to solve problems,

(ii) encourage students to think of appropriate metacognitive strategies for problem solving,
motivate students to monitor the use of these metacognitive strategies during problem solving, and

boost students’ performance on the topic in hand.

In order to test these hypotheses we needed to implement the model within a problem solving ILE. Instead of using an existing environment, we decided to build an ILE for algebra word problems and incorporate the RA Model in it. We called this system MIRA (Metacognitive Instruction using a Reflection Approach). All the activities, data repositories, and inference engine proposed in the RA were implemented.

To investigate the power of this new integrated environment (MIRA with RA) we conducted an experiment. A controlled study was devised and the data it provided was used to draw conclusions about the effectiveness of the RA Model for metacognition training.

4.6 Conclusion

In this chapter we have presented the Reflection Assistant model, a generic computational model for metacognitive instruction in problem solving learning environments. The architecture of the RA was presented and details about its components were discussed.

The ultimate goal is to create a comprehensive problem solving environment that provides activities that serve to anchor new concepts into the learner’s existing cognitive knowledge to make them retrievable. We believe that the metacognitive training proposed by the Reflection Assistant model can help with that.

One important innovation introduced by the RA Model is the idea that it is necessary to create purposefully and separate situations where self-reflection can occur. In other words, that it is necessary to conceive specific moments and activities to promote awareness of aspects of the problem solving process. Therefore, the RA is designed as a separate component from the problem solving environment with its own specialized activities.

Moreover, two new stages in problem solving activity are proposed: pre-task reflection stage and post-task reflection stage. During these stages the student interacts solely with the reflective activities proposed in the RA Model, which focus on her metacognitive skills and her problem solving experience.

The RA Model follows principles of reflective tools as facilitators, i.e. tools that aim to help student becoming self-regulators of their own learning (Reusser, 1993; Collins and Brown, 1988). As such, the RA adopts the following instructional principles:

1. The RA intends to promote students’ sense of autonomous thinking. Therefore, it is not directive and it does not tell the learner what to do. It suggests, encourages, and reifies students actions.

2. The RA provides feedback on strategies used, actions taken, and performance from the metacognitive perspective.

3. The RA tries to avoid cognitive overload. It lets students perform their problem solving activity with minimal interference during the production phase.
4. The RA coaches the metacognitive skills overtly and explicitly, but always contextualized in the domain; keeping the balance between domain general activities and task activities is essential to the success of the metacognitive training.

As the RA is a generic model for problem solving environments its activities and components have to be adjusted and implemented according to the specific domains and environments it will be used with. Prior to implementing the RA for a specific learning environment the instructional designer has to review the reflective activities proposed and adapt their language to the domain at hand. Besides that, the programmer has to implement the modules and data repositories of the RA. Additionally, new data repositories may have to be included in the problem solving learning environment, which will capture the history of the students’ interactions and problem solving performance. These data are used by the RA, following the information provided in table 4.5. In Section 4.4.6 we have presented the communication points between the RA and the learning environment in detail.

In order to test the RA model we have implemented it into a new learning environment for algebra word problems, called the MIRA System. So, we had to implement both the environment and the RA modules and data repositories.

The next chapter presents the domain of algebra word problems used in the MIRA system. After that, the other two chapters are dedicated to show the design and functionality of MIRA, and how MIRA incorporates the RA Model.