A System for Concerned Teaching of Musical Aural Skills

Geraint A. Wiggins*
Department of Computing
City University, London
Northampton Square,
London, EC1V 0HB
England
geraint@city.ac.uk

Shari Trewin*
Division of Informatics
University of Edinburgh
80 South Bridge
Edinburgh EH1 1HN
Scotland
shari@dai.ed.ac.uk

Abstract

We present *teaching concerns*, a domain-independent mechanism for controlling the content and form of automated teaching interactions. Concerns represent commonsense teaching requirements such as challenging a student. Concerns have been implemented in **Tapper**, a new system for musical ear training. We describe and give real examples of concerns in action within **Tapper**. We claim that this approach produces sensible and explicable individualised interactions within a loosely structured curriculum.

1 Introduction and Overview

1.1 Motivation

In this paper, we describe the teaching strategy applied in **Tapper**, a new system for musical ear training, and give examples of teaching in progress in experimental sessions with children.

A particular contribution of this work is the inclusion in **Tapper**'s control mechanism of the notion of *teaching concern*. A teaching concern is a particular governing aspect of the teaching process, on which a teacher might be focussed at a particular time. For example, teaching concerns implemented in the current system include boosting the confidence of a beginner and broadening the coverage of the student's knowledge. Not all concerns are active at all times, and their changing strengths interact to control the behaviour of the system. **Tapper** chooses a path through the curriculum based on the teaching concerns and on the performance of the student, giving a more natural and precisely tailored learning experience than would be otherwise possible.

1.2 Overview of the Paper

In §2, we survey related ITS technology. In §3, we give a high-level overview of the architecture of **Tapper**, then focussing on the primary topic of this paper, the use of *teaching concerns*, in §4. We present some case studies of concerns in action in a formative evaluation of the system in §5. Finally, in §6 we summarise the contribution of the work, and suggest avenues for further research.

2 Related Work

Our work is set in context of several well-known ITS architectures, all of which encode a notion of *teaching strategy*, though none of these is exactly the same as that in **Tapper**.

[1]This work was mostly carried out while the first author was working at the Division of Informatics, University of Edinburgh. The second author is now working in the Accessibility Research group, IBM T J Watson Research Center. Correspondence should be addressed to the first author.

Sokolonicki (1991) gives a good survey of ITS architectures; there is little point in reiterating his work here. In summary, many existing ITSs support reasoning systems of some kind which guide the teaching process in terms of *strategy*. In most cases in this survey, a strategy is a high-level decision to which an ITS strictly adheres, often for an extended period. For example, the decision to use the Socratic method, or choosing the balance between teaching and assessment, is the level of strategy discussed. However, Major (1995) cites evidence that in human teaching situations, strategy is also continually reassessed at a lower level, adjusting individual aspects of a lesson when appropriate, so that "a complete strategy change would happen gradually in a series of smaller steps" (p. 121). This view is corroborated by our education consultant, Joanne Armstrong (personal communication).

DOMINIE (Spenseley *et al.*, 1990) also features explicit reasoning for teaching strategy selection. It has strategies at the level of (*e.g.,*) "cognitive apprenticeship" and "practice". DOMINIE makes its strategic decisions according to several heuristic factors, such as "achieving a balance between teaching and assessment". These heuristic factors do not, as they stand, encode a teaching behaviour – rather, they ensure that whatever strategy is chosen is used coherently. DOMINIE was the first system to automate selection from several different strategies, and Spensley *et al.* note that "It is not clear what theoretical basis exists for this choice because current teaching practices do not often involve extended one to one interactions" (p. 199). Our work addresses this issue: musical ear training is an example of one to one interaction; and we are studying the emergent properties of just such a multi-objective strategy selection procedure.

GTE (van Marcke, 1998) and PEPE (Wasson, 1998) also reason about instructional strategy. Further, Van Marke, quoted by Wasson (p. 301), notes that teachers' "instructional knowledge" allows them to "adapt their teaching strategy according to the situation", and both systems are very knowledge-rich in this respect. However, as with DOMINIE, their knowledge is expressed in terms of how to teach a given curriculum, or particular aspects of it, rather than in terms of an over-arching, on-going detailed analysis of the student's behaviour within that curriculum.

The work to which ours owes most is that of Major (1995), who has designed an ITS construction system, COCA. COCA constructs multiple-strategy systems, and the user specifies the strategy selection reasoning. COCA-built systems have a fixed operating cycle: decide which concept to teach next; decide which tutorial activity and style to use; choose the content of the next interaction; and act on student assessment. COCA has two levels of strategy, high, as above, and low level, which decides what kind of concept (*e.g.,* easier, harder or new) to proceed to next. COCA uses a set of rules to make these decisions, which are hard-wired by the ITS designer. The rules may encode heuristics, but apparently do not have a mutually coordinated response to the current situation: they respond only disparately to data in the student model, and there is no means of describing a unified response mechanism. It is this gap that the present work aims to fill.

3 Architecture

To set the context in which concerns are applied, we describe the architecture of **Tapper**, as illustrated in Figure 1. The blocks in the diagram are explained below. In a paper of this length, we cannot give the full detail of a complex system such as this; readers needing more detail are referred to (Trewin *et al.*, prep). However, this overview is enough for the reader to understand §4, which introduces our central topic: using *teaching concerns* to direct the student's course of study.

3.1 Course Specification

The *course specification* is a logical description of the material to be taught in the ear training course. Each section of the material is associated with an indication of which *methods* (see §3.2) the system could use to teach it, and how the material maps on to the variables in the method.

The curriculum is specified in terms of *topics*, *examples* and *questions* to use in teaching the topics. The current curriculum, biassed towards musical ear training, consists of topics such as "pulse", "beats in a bar" and "rhythm". Examples are frames which will be instantiated by the system with one of a set of musical examples of equivalent difficulty.

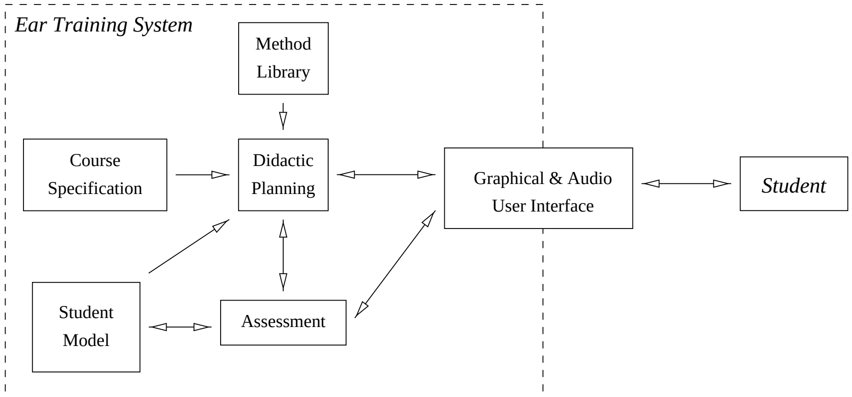

Figure 1: The Architecture of the Experimental System, **Tapper**

Each topic is uniquely named, and has a difficulty level and a set of methods appropriate for teaching it, ordered by difficulty. Each topic also has *pre-requisites*, which must be achieved by the student before it is first introduced.

3.2 Method Library

The *method library* is a logical description of *methods* by which material may be taught, specified in the abstract. The curriculum contains no information about how to present the material to students; presentations are generated on the fly. This avoids the need to encode individual lessons for each topic: one simply specifies which methods are appropriate and the system does the rest.

Eleven methods are available in the current version, some general and some ear-training-specific. They include: demonstrating a topic; asking different kinds of question; and asking the student to improvise to fill in a gap in an example.

Each method specifies the class of teaching examples for which it is appropriate. Most examples fit into several classes.

New methods may be added to the system in a standard way; so long as they conform to **Tapper**'s representation, they will be integrated automatically with existing methods and curricula.

3.3 Student Model

The *student model* records information about the student's progress through the curriculum. It provides information for other modules in the system to use when making decisions. Other modules can query the student's standard or exposure to specific elements in the course specification, or their overall performance.

Students are also assigned a specific skill level (1-3) according to their performance. Skill levels are used by the *assessment module* (see §3.5) and the *didactic planner*.

3.4 Didactic Planning

The highest level of control in **Tapper** rests with the *didactic planner*. Having loaded a student profile, or started a new one, it executes sequences of cycles, like those in COCA (Major, 1995). Each cycle consists of presenting material to the student, gathering their response, analysing it, providing feedback, and then updating the student model and session records.

As in COCA, **Tapper**'s overall behaviour is governed by a high-level strategy. In **Tapper**, the strategy module makes broad and fairly simplistic suggestions, for example suggesting a change when things are getting too easy for the student, or repeating a cycle when a student has done very poorly. When these strategies do not fully specify the content (topic, method and example) of the next cycle, *teaching concerns* are used to make the necessary decisions. These concerns interact with the low-level strategies to produce a behaviour which is coordinated, but which is potentially more responsive to the needs of the particular student than is possible in a domain-independent system without multi-level strategy. This interaction is explained in §4.

3.5 Assessment

The *assessment module* is an inference system which uses information from the student and the course specification to assess the student's knowledge, in ways specified by library methods. The assessment of the student's musical input is by no means trivial, but will be discussed elsewhere (Trewin *et al.*, prep).

3.6 Graphical & Audio User interface

The user interface of the system is currently geared towards the application with which we have been experimenting: musical ear training. It is capable of displaying music in various ways, both visually and aurally, and is able to read input from the student in the form of text typed on the computer keyboard and rhythms tapped out on one of several input devices.

4 Using Concerns to Guide Teaching

4.1 The Concern Mechanism in Tapper

Tapper's high level teaching strategies do not usually fully dictate the contents of the next interaction. **Tapper** also uses a set of *teaching concerns* to make the necessary decisions, sometimes overriding the high level strategy altogether.

There are four primary teaching concerns in the current prototype: boosting student confidence; challenging the student; reinforcing material already taught; and increasing coverage of the curriculum. More concerns could be easily added in future.

All concerns are assigned numerical values. 0 indicates that **Tapper** is unconcerned about the property in question. The maximum concern in **Tapper** is currently 7, though this is arbitrary.

In the current prototype, teaching concerns are initialised by examining five variables:

- the student's performance in the previous cycle of this session;
- the average of the student's performances in the previous four cycles;
- the current activity (try a topic for the first time, practice, recap or revise);
- the student's performance with the current topic, if a topic is defined; and
- the student's overall performance for all cycles attempted.

For each concern, a set of *influences* is defined in terms of the above variables. Each influence has a positive or negative strength in the range 1-3. For example, if the average of the student's recent performances is very poor, this has a positive influence of strength 3 on concern for boosting the student's confidence, and a negative influence of strength 3 on concern for increasing coverage within the curriculum. The influences of all the variables on each concern are combined by summing the total influences and limiting the result to the permissible range.

Concerns are used identically for all **Tapper**'s main didactic planning, whether choosing topics, methods or examples. The mechanism is domain- and curriculum-independent, and allows addition of new topics, methods or examples to the curriculum, so long as the existing curriculum representation is adhered to. Concerns are used to choose appropriate *low level strategies* for choosing the contents of the next cycle. Existing low level strategies include:

- choose something of similar/higher/lower difficulty to/than the last one;
- choose something easy/hard/average;
- choose something the student is strong/average/weak at;

- reuse the previous choice;
- go to something which builds on the last choice;
- choose something new to the student;

Each low level strategy is related to the four teaching concerns, either being a good strategy to address the concern, a reasonable strategy, indifferent, or bad. For example, the strategies of choosing a new or harder topic, or a topic the user is weak on, are good for addressing concern about challenging the user. Strategies which choose easy topics, or topics the user is good at, are bad strategies for that concern.

All of the available low level strategies are scored according to the current set of concerns, ordered, best first, and tried, in order, until a solution is found. Several strategies may be tried, since not all strategies are applicable in every situation. For example, the strategy *harder* cannot be applied when the student is already on the hardest option.

When updating the teaching concerns after choosing one aspect of the next interaction, **Tapper** decides how well each concern has been addressed by the strategy applied. If it has been addressed, its strength will be reduced. At the same time, new concerns will be introduced, if appropriate.

The rest of this section uses a real interaction sequence to illustrate how teaching concerns and low level strategies are used to choose the contents of an interaction.

4.2 Example: A Student Performing Well

In this example, a capable student, L, is having her first lesson with Tapper. She is studying her second topic: tapping the accented beat of a tune, and has performed three cycles well on this topic. Throughout the session, she has used a single, easy method, and easy examples. At this stage, high level teaching strategy dictates that she should be given more practice with the current topic, but should move on to a new method. The teaching concerns are used to choose the method and then the example for the next cycle.

L's recent performance has been rated as 'good', her performance in the previous cycle was 'very good', her overall performance is 'good', her performance on the current topic is 'good', and her activity is 'practice'. Using this information, **Tapper** examines the influences for each concern. L's recent good performance has a negative influence on concern for boosting her confidence, level 1, so the overall level of concern for boosting confidence is limited from -1 to 0. None of the influencing variables affects concern for increasing coverage of the curriculum, or for reinforcing the material already taught, so those concerns are also set to 0. L's overall good performance has a positive (level 1) influence on concern for challenging her, so that concern total is 1.

Given the single concern of challenging the student, the system orders the set of known strategies as shown below, from the strategy most likely to choose a challenging method, to the least likely.

- weak – choose a method the student is weak on;
- harder – choose a harder method than the previous one;
- average – choose a method the student is average at;
- different – choose a different method to the last one;
- next – choose the next method (a suggested order in which to use methods is given for each topic, and a general default ordering is also defined);
- hard – choose a hard method;
- previous – choose a method used before the most recent;
- strong – choose a method the student is good at;
- easier – choose an easier method;
- easy – choose an easy method.

So **Tapper** first tries to find a method L is weak on. This fails, since she has only tried one method, and is doing well with it. **Tapper** then tries to choose a method harder than the current one. Since a topic has already been chosen, it does so by referring to the ordering for methods provided for that topic in the curriculum. In this way, it finds a more difficult method: *fillin*, in which the student is asked to perform without having heard the example in advance.

Having chosen a method, an example is needed. First, **Tapper** adjusts the set of concerns to take into account the method chosen. The strategy *harder* is good for addressing concern about challenging the student. So, the level of concern for challenging the student is reduced, and returns to 0.

At this stage, **Tapper** has no particular teaching concerns, so the default strategy ordering is used to choose an example. The first strategy tried is *same* – use the same kind of example as the previous cycle. In this case, the kinds of example required by the two methods are identical, so the strategy succeeds, and chooses an example of the same form and difficulty as the previous cycle (though not the same tune), and so the session proceeds.

5 Concerned Teaching in Action

Formative evaluation of the **Tapper** system as a whole has been carried out in a school environment with four eight year old and four twelve year old children. The children used **Tapper** for three 20 minute sessions, each subsequent session building on progress made previously.

The children responded positively to **Tapper**, and found it easy to use. They experienced some difficulty with some parts of the curriculum, prompting revision of the demonstrations and explanations provided in the original curriculum. Further development of the feedback given on each cycle is also in progress, in consultation with the children's own music teacher, who acted as consultant on this project. This work will be reported elsewhere. The rest of this section gives two example sessions recorded in this evaluation, focussing on the way in which concerns interact with the higher level teaching strategies to control the contents of each cycle. Less interesting parts of the sessions are abridged.

5.1 Case Study 1: A First Session

This example is the first session for G_1, a boy aged 12. G_1 had been playing the piano for 3 years, also played the guitar, violin and chanter, and was of average musical ability. At the end of the session, he reported that he thought **Tapper** was a fun, low pressure way to learn rhythm, though he would have liked fuller explanation of some of the topics and methods. Although initially quite unsure, by the end of the session he felt quite confident about being able to do the exercises.

In this case study, we are starting from the very beginning, so there are some standard concerns for beginners: boosting confidence and broadening coverage. A topic must necessarily be chosen, and then demonstrated to the student; the student then responds with an indication that he is ready to continue. In the first cycle, an easy topic is chosen and demonstrated. Output from the system for the second cycle follows, edited for readability.

```
Cycle 2:

High level teaching strategies:  Continue with current topic, different
    method.

Choose a method:
    Concerned with boosting the student's confidence at level 5.
    Strategy "average" failed.
    Strategy "strong" failed.
    Strategy "easier" failed.
    Strategy "easy" succeeded.  Choice made: methodName(followcue)

Choose an example:
    No concerns.
    Strategy "same" succeeded.  Choice made: pulseFrame(easy)

The student responded, but the result was very poor, scoring 1.
```

Unfortunately, G_1 has the wrong idea of how to respond to the exercise, and has scored a very low mark. While detection of such situations would be an interesting extension to **Tapper**, it cannot currently distinguish between different kinds of poor performance, so it simply tries the same topic and method again. However, the *same* strategy will choose a different example from the same set, to keep things interesting. Because of G_1's poor performance, the confidence and coverage concerns have risen to maximum level.

```
Cycle 3:

High level teaching strategies:  Repeat topic and method.

Choose an example:
    Concerned with boosting the student's confidence at level 7.
    Concerned with reinforcing what has already been learned at level 7.
    Strategy "average" failed.
    Strategy "same" succeeded.  Choice made: pulseFrame(easy)

The student responded, and the result was good, scoring 83.
```

This time, G_1 has got the idea, and performs well on the example. He continues to do so for the next two cycles, scoring 62 and 79, until the high level strategy suggests a change of method to avoid boredom. Now, the teaching concerns begin to make a significant difference.

In choosing the next method, **Tapper** takes into account the same two concerns, with a bias towards reinforcement, and generates a list of strategies to try. The concern levels are based partly on the mixed results of the preceding exercises. The *average* strategy (see end of §4.2) addresses both concerns, while *weak* compromises in favour of reinforcing. *Previous* and *easier* are good for boosting confidence. However, none of these strategies succeeds, because, at this early stage in the teaching process, there are no appropriate options to take. Finally, **Tapper** tries to find an *easy* method, to boost confidence, and succeeds. However, the method it has chosen is the same as before, so the well-informed concerns have overridden the high level, but ill-informed, strategy of choosing a new method. Recall that the *same* strategy chooses another example from the same set, not just the same example.

```
Cycle 6:

High level teaching strategies:  Repeat topic, change method to avoid
    boredom.

Choose a method:
    Concerned with boosting the student's confidence at level 4.
    Concerned with reinforcing what has already been learned at level 6.
    Strategy "average" failed.
    Strategy "weak" failed.
    Strategy "previous" failed.
    Strategy "easier" failed.
    Strategy "easy" succeeded.  Choice made: methodName(followcue)

Choose an example:
    Concerned with reinforcing what has already been learned at level 6.
    Strategy "same" succeeded.  Choice made: pulseFrame(easy)

The student responded, and the result was very good, scoring 98.
```

This selection has addressed the concern about boosting confidence, so its level is reduced. However, the *easy* choice of method does not reinforce, so that concern is unchanged. The high level strategy is still to change method, as this was not achieved last cycle. However, the student has not yet demonstrated a level of ability at which a change is possible, so cycle 6 is repeated at cycle 7. G_1 scores 63.

In cycle 8, the high level teaching strategy is still suggesting change, this time in topic as well as method. The current student model is not creating concerns, and so the strategies are tried in the default order; therefore, the same topic is chosen again. However, this different ordering of strategies gives a different choice of method – a harder one, and it shows in the score: 39. Cycle 9 is therefore made to repeat cycle 8, and this time G_1 scores 77.

Cycle 10 sees the high level strategy again overridden by concerns. **Tapper** tries to find a topic which will reinforce what is already learned, but the best possibility turns out to be the topic which has just been covered. So it chooses that same basic topic again. G_1 scores 100.

```
Cycle 10:

High level teaching strategies:  Change topic to avoid boredom.
Choose a topic:
    Concerned with reinforcing what has already been learned at level 2.
    Strategy "weak" failed.
    Strategy "average" succeeded.  Choice made: topicName(pulse)

Stick with the same method: methodName(fillin).
```

```
Choose an example:
    No concerns.
    Strategy "same" succeeded.  Choice made: pulseFrame(easy)

The student responded, and the result was very good, scoring 100.
```

Now, G_1 has succeeded enough to cause a new concern: to challenge him. This leads to the choice of a new topic. After trying and failing to find a topic that G_1 is *weak* at, **Tapper** finds a topic which is *harder* than the current one, and selects it. If G_1 had not succeeded at the *pulse* topic, this option would not apply, and the choice would default to *pulse* again. In a more diverse curriculum than that of our prototype, this repetitious behaviour would not arise.

We now move on to another student who is more advanced in her work with **Tapper**.

5.2 Case Study 2: A Second Session

Our second case study is taken from the second session with **Tapper** for G_2, a girl of 12 years old. G_2 was a capable musician and **Tapper** responded to herwell. She passed the easy sections quickly, and soon reached areas where **Tapper**'s advanced features can clearly be seen.

G_2 had been playing the violin for four years. She was a competent musician. She reported that, although she had found a problem with the user interface, she enjoyed working with **Tapper** (particularly the way the exercises got harder with time – see below), and that felt that she had made progress.

With all the subjects' second sessions, teaching follows on from the first session, as recorded in the student model. We join this session after cycle 3; the current topic is *accented beats*. G_2 is doing well, so current concerns are with challenging her and broadening her coverage of the curriculum. Method-choosing strategy *new* addresses both these concerns, but there is no appropriate new method, so **Tapper** chooses a *harder* method.

```
Cycle 4:

High level teaching strategies: Stick with the same topic,
    change method to avoid boredom.

Choose a method:
    Concerned with challenging the student at level 6.
    Concerned with broadening the student's coverage at level 3.
    Strategy "new" failed.
    Strategy "harder" succeeded. Choice made: methodName(fillin)

Choose an example:
    No concerns.
    Strategy "same" succeeded. Choice made: accentFrame(easy)

The student responded, and the result was very good, scoring 94.
```

Following this strong score, high level strategy is to stick with the same topic and method in cycle 5, but concerns are still to challenge and broaden. A harder example is chosen, and this time G_2 does less well, scoring only 65. The resulting concerns in cycle 6 are changed: the challenge has gone, and the level of concern for broadening is reduced.

Broadening coverage is less appropriate to examples than to topics and methods, but the selection here is still logical – **Tapper** has chosen a harder exercise to broaden the student's ability within the set of examples appropriate for that topic and method.

```
Cycle 6:

High level teaching strategies: stick with the same topic and
    method, but choose a new example.

Choose an example.
    Concerned with broadening the student's coverage at level 1.
    Strategy "new" failed.
    Strategy "next" failed.
    Strategy "harder" succeeded. Choice made: accentFrame(hard)
```

```
The student responded, and the result was very good, scoring 96.
```

Cycle 7 sees the concerns being assessed twice, once for a change of method, and once for a change of example. Here, the concerns are supporting the high level strategy, and would do so even more effectively with a fuller method set, where a high-scoring strategy (*e.g., new*) would have been successful. However, *different*, though normally a good strategy for challenging, can only find a method which is in fact easier than the current one.

```
Cycle 7:

High level teaching strategies: Stick with the same topic,
    change method and example to avoid boredom.

Choose a method.
    Concerned with challenging the student at level 2.
    Concerned with broadening the student's coverage at level 1.
    Strategy "new" failed.
    Strategy "harder" failed.
    Strategy "next" failed.
    Strategy "weak" failed.
    Strategy "average" failed.
    Strategy "different" succeeded.
        Choice made: methodName(followcue)

Choose an example:
    Concerned with broadening the student's coverage at level 1.
    Strategy "new" failed.
    Strategy "next" failed.
    Strategy "harder" failed.
    Strategy "same" succeeded. Choice made: accentFrame(hard)

The student responded, and the result was very good, scoring 100.
```

In the final cycle of this case study, G_2's excellent score in cycle 7 has led again to concern for challenge and broadening. The high level strategy is anyway to change topic, so **Tapper**'s components are unanimous in trying the *new* strategy. G_2 has already met the prerequisites of a new topic, and this is selected and demonstrated.

5.3 Discussion

The two case studies demonstrate the effect of the concerns on the teaching process, and how they respond to the student's performance. In particular, two points should be noted.

First, there is a dynamic balance between the concerns of boosting confidence and of challenging: G_1's case study starts with the aim of reassuring, and begins to bias towards the more demanding concerns as G_1 becomes able to cope.

Second, **Tapper**'s low-level response to the concerns can produce a very reasonable emergent behaviour over time, as exemplified in cycles 5-7 of G_2's case study, where the material taught gets progressively – but smoothly – harder as G_2 learns.

This behaviour arises exclusively from the concerns and their effect on the low-level strategies and is not hard-coded into the system. As such, it is very general and readily extensible.

Also, the studies show several ways in which **Tapper** could be improved, notably in its ability to understand the nature of errors, and in the coverage of its curriculum, whose poverty restricts the power of the concern mechanism to respond to the user.

6 Conclusion and Further Work

This project has shown that the emergent behaviour of an ITS, based not on pre-programmed transition through a curriculum, but rather on a mixture of high- and low-level localised reasoning, can lead to a natural and precisely tailored learning experience for each student. In this paper, we have demonstrated this behaviour *via* three examples of our system's operation.

Note that the mechanism we have described is domain- and high-level strategy-independent, and so may be expected to generalise to other areas of education seamlessly. This means that the architecture on which **Tapper** is based can form an effective framework for comparing different aspects of teaching systems, to a high degree of granularity, because of the independence of its various modules.

A particularly important aspect of this work is that real teachers tend to think in terms like our concerns. This means that a system based on this way of thinking is more likely to be amenable to use by teachers for contstructing their own ITSs (though **Tapper** in its current form is not appropriate for this).

We expect that this work will continue in several directions: extending the curriculum; building better high-level strategies; improving the understanding of student input; and increasing the range of modalities by which the student can interact with **Tapper**.

In future versions of **Tapper**, we anticipate that the basic strategy would be extended with more sophisticated strategies. For example, when a student has performed badly on a long example, an appropriate strategy would be to split the example into parts, teach each part separately, then reconstruct the original example.

Finally, further experimental work is required, to verify **Tapper**'s response to a wider range of children of different musical backgrounds.

Acknowledgements

This work was supported by UK Engineering and Physical Sciences Research Council research grant number GR/L99029 and by City University, London.

We are enormously grateful to Joanne Armstrong, of Fettes College Junior School, Edinburgh, who was educational consultant to this project, and who designed the educational content of **Tapper**'s curriculum and methods. We also thank Luke Phillips, who helped with the evaluation of student input, and Márcio Brandão, who, with Luke, helped to run the experiments and to process the results.

References

Major, N. (1995). Modelling teaching strategies. *Journal of Artificial Intelligence in Education*, **6**(2/3), 117–152.

Sokolonicki, T. (1991). Towards knowledge-based tutors: a survey and appraisal of intelligent tutoring systems. *Knowledge Engineering Review*, **6**(2), 59–95.

Spenseley, F., Elsom-Cook, M., Byerley, P., Brooks, P., Federici, M., and Scaroni, C. (1990). *Using multiple teaching strategies*, pages 188–205. Ablex Publishing Corp., Norwood, New Jersey.

Trewin, S., Armstrong, J., Phillips, L., and Wiggins, G. A. (prep). An intelligent teaching system supporting music education research. In preparation.

van Marcke, K. (1998). GTE: An epistemological approach to instructional modelling. *Instructional Science*, **26**(3/4), 147–191.

Wasson, B. (1998). Facilitating dynamic pedagogical decision making: PEPE and GTE. *Instructional Science*, **26**, 299–316.