Task and Interaction Regulation in Controlling a Traffic Simulation

Patrick Jermann

Task and Interaction Regulation in Controlling a Traffic Simulation

Patrick Jermann

TECFA - University of Geneva
Patrick.Jermann@tecfa.unige.ch

ABSTRACT

In collaborative problem solving, metacognition not only covers strategic reasoning related to the task but also reasoning related to the interaction itself. The hypothesis underlying this work states that regulation of the interaction and regulation of the task are closely related mechanisms and that their co-occurrence facilitates collaborative problem solving. These assumptions are tested experimentally with a traffic simulator. The results show that co-occurrence of task and interaction regulation allow quicker solving of the problem, thus better performance. The experimental treatment aims at observing the effects of interaction meters on the accuracy of subjects' estimation of their participation. Interaction meters are visualization tools that represent the number of contributions related to the discussion and to the implementation of the solution.

Keywords
Metacognition, task regulation, interaction regulation, reflective tools, interaction meters

INTRODUCTION

When we solve a problem with a partner, we usually discuss a plan before starting to act and make up new plans as the activity goes on. These plans might not only define the actions to be performed, but might also define the roles of each participant. In this paper we propose to examine the conditions in which determining "what to do" is accompanied by a reflection on "who does what". In other terms, we will examine if it is possible to foster the co-occurrence of task regulation and interaction regulation. We also will examine the effects of co-occurrence of task regulation and interaction regulation on performance.

The goal of our work is twofold. From a CSCL system designer viewpoint we aim at demonstrating the efficiency of interaction meters, which are visualization tools that functions like a mirror of the activity's organization. We thereby try to establish a link between initial conditions (the interaction meters as characteristics of the tool) and the characteristics of the interaction. From a cognitive psychology viewpoint, we attempt to describe the functioning of a sociocognitive system by adopting the distributed cognition perspective (Salomon, 1993). We try to establish a link between characteristics of interaction (co-occurrence of task and interaction regulation) and performance. Three factors intervene in this framework: the initial conditions, the interaction and the products of activity. In this now widespread paradigm, interaction acquires a double status, as dependent and independent variable (Dillenbourg, Baker, OMalley & Blaye, 1995).

From initial conditions to the characteristics of the interaction

In order to address the question of the conditions that foster integration of task and interaction regulation we designed a tool that functions like a mirror for the pair's activity. This corresponds to one of two broad approaches to support collaboration. The first approach consists in structuring the situation in which the collaboration takes place. Factors like group composition, technological setting and nature of task allow orienting the nature of the interaction that will take place. The second approach, we chose for our current purpose, consists in coaching and regulating the interaction as it unfolds. We previously proposed a framework to categorize systems that implement the coaching approach (Jermann, Soller & Muelhenbrock, 2001). The categorization is inspired by Barros and Verdejo's work (2000) and relies on a simple model that describes coaching as a homeostatic process. The process starts with a phase of data collection about the ongoing interaction. The raw data is then transformed into a set of indicators that express an underlying model of interaction (e.g. a basic model could contain the ratio of talk by problem –solving actions as an indicator of planning activity). When discrepancies between the current state of interaction and the desired state are observed, remedial actions are undertaken. Different actors can carry out the diagnosis and remedial actions. On one hand, the teacher or the learners themselves could maintain a model of interaction and detect inconsistencies. On the other hand, a computer could help keeping track of actions and update an internal computational representation of the state of the interaction. Some systems contain a module that can give advice based on the analysis of discrepancies between the current state and desired state of the interaction.

The current state of the interaction might also simply be displayed to the teacher or the learners as a graphical representation. The interaction meters we used in our research are an example thereof. Subjects are presented a constantly updated visualization of their participation while they solve the task. The visualization represents participation in talk and task related actions through bar charts that show the number of messages and the number of problem solving actions. Interaction meters take over the data collection and summarization phases of the coaching process described above by making participation and work organization explicit. The design rationale of interaction meters is that they may raise problem-solvers' awareness of participation and work organization and thereby help them regulate the interaction.

We designed two different interactions meters. The first version compares subjects by representing their participation side by side as two bars. The second version represents participation cumulated across subjects, i.e. one bar chart represents the sum of the subjects' contribution to discussion and another bar chart represents the sum of the subjects' problem solving actions (see figure 3). While the first version might give subjects a more accurate estimation of their respective participation because it displays them side-by-side, the second version closer corresponds to the distributed cognition metaphor of two persons constituting a single entity.

From interaction characteristics to outcomes

Task regulation is one aspect of metacognition as defined by Schoenfeld (1987) and Brown (1987). It includes knowledge about one's own cognitive processes, action regulation and control, as well as intuitions and conceptions about the context the activity takes place in. We focus here on the second aspect, namely action regulation and control. This includes planning forthcoming actions, supervising current activities and evaluating past actions. In its initial formulation, metacognition is described as a mechanism in individual problem solving. Does the concept of metacognition scale up to a group of individuals who solve a problem? Nickerson (1993) asks this question in the context of the distributed cognition approach (Salomon, 1993). This approach considers a group of persons and the tools they use as a single cognitive system and the question is if such a system also has metacognitive skills.

Collaborative problem solving consists of two activities, solving the problem on one hand, and collaborating on the other hand. We propose that in this context, metacognition not only covers reasoning related to the task but also reasoning related to the interaction itself. We designate these activities by the terms task regulation and interaction regulation. Task regulation consists of elaborating a problem solving strategy, evaluating and planning actions. Interaction regulation consists of organizing work inside a group by defining roles or defining and assigning sub-tasks to participants. We make the general hypothesis that simultaneous regulation of work and interaction is more efficient than regulation of the task alone because it leads to a better coordination of actions.

This paper is organized in three sections. We will first describe the task, the tool and the experimental design of the study in the Method section. Then we will report and discuss results from an experimental study, finally we conclude with suggestions for future work.

METHOD

Task

The task consists of tuning a traffic-light simulation. Pairs of subjects have to tune the green-red periods of traffic lights in a way to minimize the cars' waiting time at intersections. A shared simulation presents the traffic situation and can be acted upon by the two subjects simultaneously. The actions of one subject are immediately visible on her colleagues' screen.

The subjects are asked to tune a traffic situation that contains four intersections (A, B, C and D in figure 1). Each intersection contains two lights that control the traffic on the horizontal and vertical lanes. Cars are traveling straight and never make turns, although this could be a possibility for further experiments.

Tuning an intersection requires the modification of three parameters, the length of the phase, the proportion of green and the offset. Figure 2 shows a snapshot of a tuning slider used by the subjects to modify the parameters of intersections. The optimal tunings of such a system can be determined through techniques like linear programming, but observing traffic flows and making successive adjustments by following three simple rules leads to good results as well. These rules were presented to subjects during a tutorial session preceding the experiment. They were purposely left vague in order to let subjects come up with an operational method.

Rule 1: The phase length has to be related to the total number of cars crossing the intersection.

The length of the phase equals the sum of the green and orange times for the two lights and has to be proportional to the total amount of cars that cross the intersection. This is the most critical parameter because, even if the proportions of green time match the proportions of incoming flows, a length too short corresponds to an under-

capacity of the intersection, resulting in an ever-increasing queue length at the intersection. On the other hand, and quite counter-intuitively as it appears from the subjects' tunings, if the phase length is too big, time gets wasted.

Figure 1: Traffic pattern and close-up of intersection B. Left: the horizontal and vertical lines represent lanes. The thickness of the lines represents the size of the traffic flow on each lane. Right: the small boxes on the lanes represent cars. The circles represent traffic lights. The color of the circles changes from red to green, to orange. In this snapshot, light 3 just turned green and cars start to cross the intersection.
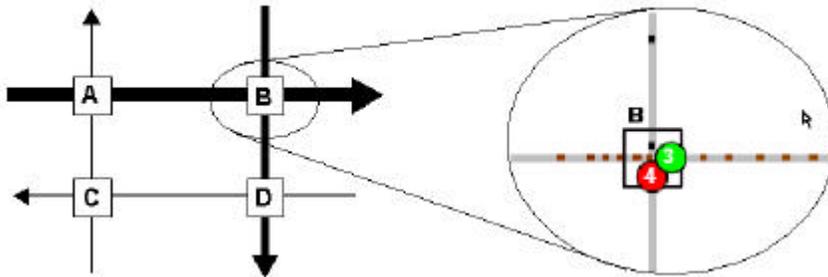

Rule 2: The proportion of green time allocated to a light has to be related to the number of cars entering the intersection through this light.

The proportion of green splits the phase in two parts and allows setting the proportion of the phase length allocated to each light. For example, if 1/3 of the total traffic crossing intersection B comes from the north (through light 4) and 2/3 come from the west (through light 3), the proportion of green should match the proportion of traffic flows. For a cycle length of 60 seconds, this means giving 20 seconds of green to light 4 and 40 seconds to light 3. It is a common behavior of subjects to accommodate a large flow of cars from one direction by increasing the phase length, even if at the same intersection a small flow of cars is allocated too much green time. This is detrimental to the overall performance of the system and it is more efficient to modify the proportion of green and give the smaller flow less time. Novices seem to pay attention mostly to large flows and neglect the smaller flows.

Rule 3: Several intersections can be synchronized by using appropriate offset values for their respective phases.

The offset is related to coordination of intersections. It allows subjects to create "green waves" of traffic. For example, a judicious use of the offset makes light 3 turn green exactly when the cars coming from intersection A arrive at intersection B.

Figure 2: Tuning slider for intersection B. The parameters can be adjusted either by repetitively clicking on the triangles or by grabbing and sliding vertical dividers (the lines separating green and red zones).
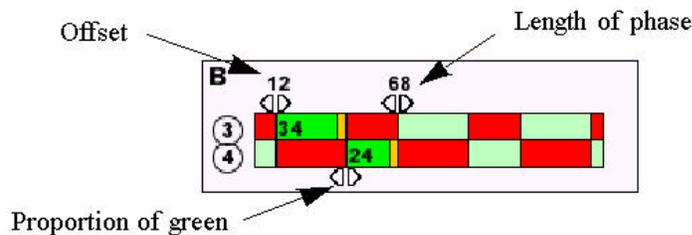
The goal of the subjects is to minimize the waiting time of cars at intersections. They can evaluate their performance by looking at a dynamic graph that shows the mean waiting time of cars traveling through the system. This graph is updated every 5 seconds. The objective of the subjects is to lower the waiting time beneath 20 seconds and keep it stable for 2 minutes. The 20 seconds limit is represented as a horizontal line on the performance graph (see figure 3 for a snapshot of the performance graph). Once this value is attained, the traffic flows change, and subjects have to adapt their solution to match a new situation.

Changes to intersections' tunings don't take immediate effect on the simulation. Parameters have to be 'sent to the simulation' by pressing a button labeled 'Update Intersections' (see figure 7 at the end of the paper). Clicking this button sends values for all the intersections to the server. The subjects are free to update the intersections as often as they want.

A 'Reset' button enables subjects to restart the simulation with clear lanes. When the traffic situation is bad, i.e. very long queues of cars at intersections, it takes long before the effect of sending better parameters are observable. To

speed up the observation of changes, subjects were instructed to use the reset button when they had the feeling that the situation was catastrophic.

The tool used in this experiment is written in JAVA and relies on a client-server architecture. Subjects communicate by using a simple chat tool integrated into the client. A snapshot of the full interface is included at the end of this paper (figure 7).

Experimental Design

The control condition and the two experimental conditions differ by the presence, in the experimental conditions, of an interaction meter next to the performance graph. The interaction meter displays bar charts representing the participation of subjects. One bar chart shows the number of words sent through the chat and the other one shows the number of problem solving action undertaken by the subjects (figure 3). The values of the bar charts are updated every minute.

In the first experimental condition called "comparative", the interaction meter displays two bars comparing the subjects' respective participation in discussion and two bars for their respective participation in tuning intersections.

In the second experimental condition called "cumulated", the interaction meter shows one bar that represents the total number of words sent by subjects, and another bar that represents the total number of tuning actions.

In the control condition, the tool only provides a performance graph and no interaction meter.

Figure 3: Performance graph and interaction meters. The line graph ('WAIT') represents performance, i.e. the mean waiting time of cars. It is updated every 5 seconds. The bar charts represent the number of words ('TALK') and the number of problem solving actions ('TUNE'). These are updated every minute. (a) Comparative condition. Subjects' participation is represented side by side. (b) Cumulated condition. The sum of subjects' contributions is represented.

(a) (b)
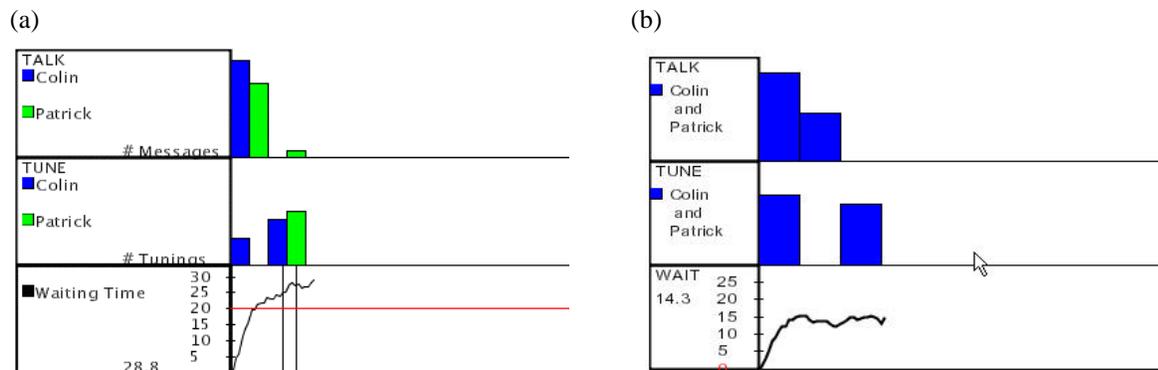
Subjects

Subjects were recruited through the subject pool associated with introductory psychology classes offered by the University of Pittsburgh. 98 undergraduate students participated in the experiment. Subjects signed up for the experiment on posters by choosing predetermined slots of time. Two time slots were available for each experimental session and the pairs resulting from this procedure were used without further consideration. The subjects were required to be native English speakers and be comfortable typing on a computer. The 49 pairs were assigned randomly to either the control condition (without interaction meter) or one of the two experimental conditions (with comparative meter, with cumulated meter).

Procedure

The experiments took place at the Learning Research and Development Center (LRDC), University of Pittsburgh. The complete duration of a session was about two hours.

Subjects were first asked to fill out a consent form. Then, during a 10 minutes session, they were given a demonstration of the tool and were told about their objective: make traffic flow as smoothly as possible through the city, bring the waiting time below the red limit. The subjects were then seated in two different rooms and were asked to go through a tutorial for 40 minutes individually. The tutorial was a walk through illustrating the effects of the different parameters that could be modified. After the tutorial, the subjects worked together during one hour with the simulation. Finally, the subjects got a post-experimental questionnaire. It aimed at assessing the representation subjects have of the interaction.

Coding

There are many coding schemes for verbal interaction with differing granularity (episodes, exchanges, turns, utterances). The goals of coding are multiple: identify the structure of conversation and relate it to learning mechanisms (Pilkington, 1999), identify levels of processing and cognitive activities (Artzt & Armour-Thomas, 1992; Amalberti & Hoc, 1998; Hoc & Amalberti, 1999; Hoc, 1999), knowledge construction mechanisms (Grusenmeyer & Trognon, 1997; Trognon, 1999; Baker, 1999; Traum, 1994), roles (Sinclair & Coulthard, 1992, in Pilkington, 1999).

Our main interest is to identify the manifestations of task regulation and interaction regulation as well as their co-occurrence. Therefore we coded the protocols at two levels of granularity. First, we coded the content of utterances to differentiate task and interaction regulation. Second, we coded episodes of planning that consist in several utterances determining a plan for action.

Table 1: Coding scheme for the content of interactions

Code	SubCode	Description
Plan		Designates utterances that refer to actions to be accomplished. They can refer to specific tuning actions (increase / decrease the length or offset of a phase, change the proportion of green for a light), or more generic activities (e.g. improve the green wave).
	Target, Parameter, Value	Each planning utterance is further described by three sub codes. The Target refers to lanes, intersections and lights ("<u>B</u> needs to be changed", "let's play with the <u>horizontals</u>". The Parameter corresponds to either the length of the phase, the proportion of green or the offset ("give light 1 more <u>green</u>"). The Value refers to the amplitude of the change planned, either qualitative ("<u>more</u> green for light 2",) or quantitative ("green as to be <u>25</u>").
Organize		Designates utterances that explicitly mention one or both subjects in relation to an action. These utterances concern the division of labor either in terms of sub-elements of the simulation (intersections, lanes) or in terms of roles (observer, implementer, measurer, …).
	Notify	Notifications about the completion of a particular sub-task. For example: "I'm done", "I changed intersection A"
	Announce	Designates utterances about the intentions and future actions of the speaker. For example: "I'll do intersection 2"
	Assign	Designates utterances about actions of the hearer. For example: "You do intersection 1"
	Ask–Grant	Those utterances are a special case of annoucement in the form of an interrogation. For example: "Can I change intersection A?". They are typically followed by an utterance that gives permission to proceed, e.g. "Go ahead !"
	Wait	For example,"Wait a second"

Sequences were defined as a group of utterances making up a plan and preceding implementation. The time separating utterances and the presence of an update action were helpful in distinguishing two consecutive planning sequences.

Table 2 and table 3 give two examples of sequences coded with the above coding scheme. In the two examples, subjects discuss about the coordination of intersections A and B. In fact, the two examples are taken from the same pair and follow each other. The first one illustrates what we mean by co-occurrence of task and interaction regulation. The second example illustrates task regulation without integrated interaction regulation.

The first example (table 2) starts by a plan proposed by S2. The offset parameter is not explicitly referred to, but we nevertheless code the utterance as containing an implicit reference to the offset parameter ("A and B work together" refers to the offset). Shortly thereafter S1 proposes a similar plan, coded Plan and Organize because S1 proposes an action and at the same time tells that she is going to perform the necessary actions. The third utterance is the response from S2 to S1's proposal. Because it is not possible to tell if S2 agrees to both the plan and the fact that S1 is going to perform the actions, we code her answer as Plan without additional arguments and with the basic Organize code. This way of coding allows us to distinguish full proposals, (i.e. Plans with a parameter and Organize with a subtype specified) from acknowledgments. After the three first utterances, S1 implements the proposed changes.

Table 2: Coding example of integrated task and interaction regulation

Code	Time	Who	Text
Plan(Target=A+B, Param=offset)	1469	S2	a and B need to work together
Plan(Target=B,Param=offset, Value=20) / Organize-Announce	1472	S1	B needs more offset time, i am going to make it 20
Plan / Organize	1472	S2	Good
	1485	S1	(Changes Offset)

The second example (table 3) start with S2 proposing a plan for lights 1 in intersection A and light 3 in intersection B. Almost at the same time, S1 makes an observation about the previous attempt to coordinate the lights. This utterance is coded Plan-Observe. In this example, subjects do not explicitly assign actions to somebody and therefore this examples does not contain Organize codes.

Table 3: Coding example of task regulation without interaction regulation

Code	Time	Who	Text
Plan(Target=1+3,Param=offset)	1543	S2	we need 1 and 3 to be green almost at the same time
Plan-Observe	1546	S1	it's still getting jammed
Plan	1557	S1	Ok
	1577	S2	(Changes Offset)
	1585	S2	(Updates Intersections)

RESULTS

After examining dialogues produced by the 49 pairs, we decided to drop 6 pairs from the analyses described hereafter because their motivation was too low. These pairs were using rude language, playing randomly with the simulation and making it obvious they were present at the experiment only to get their course credits. Before we address the hypotheses made in the introduction, we present variables describing performance, how often the interaction meters were used and how subjects organized work.

Levels of success
The goal of the subjects was to bring the average waiting time of cars below 20 seconds and maintain it below this limit for 2 minutes. Out of 43 pairs, 39.5% (N=17, referred to as 'failed') failed and 60.5% succeeded (N=26, referred to as 'success'). Due to the high percentage of successful pairs, we further distinguished between the pairs that reached the objective in less than half an hour (N=10, 23.3%, referred to as 'super') from those who succeeded in more than half an hour (N=16, 37.2%, referred to as 'normal').

Interaction meters
The performance graph as well as the interaction meters is programmed to appear when subjects move the mouse over the corresponding area. The meters disappear 10 seconds after the mouse left the area. However, as long as the subject keeps the mouse pointer within the meter, it stays displayed.

The performance graph was displayed m=77.7 (sd=29.1) times. It was visible for a total duration of m=2029.9 (sd=580.6). The mean duration spent in the performance graph is therefore 28.9 seconds (sd=13.5 sec). This means that subjects would stay in the performance graph for about 19 seconds in order to see the performance curve evolve. The performance graph was updated every 5 seconds so we can deduce that subjects were waiting for 3 or 4 new values to be displayed (plus 2 where the meter stays displayed after the subject leaves the area).

In conditions with interaction meters, the meters were displayed m=73 times (sd=35) for a mean duration of 12.2 seconds (sd=12.2). The values on the interaction meter were updated every minute, and this explains why subjects did not stay in the interaction meter long enough to wait for new values (staying in one region of the user interface for more than 60 seconds is not practical). There is no difference between the comparative and cumulated conditions when comparing the number of times subjects visited the interaction meters. However, On a 7 point lickert scale ranging from very often to very rarely, subjects in the comparative condition stated that they looked up the interaction meter more often (m=3.1, sd=1.4704) than the subjects in the cumulated condition (m=4.23, sd=1.3309) (F=9.797 p=.003).

Work organization

The organization of work is an important characteristic of the interaction. It could have an impact on performance as well as on the content of the mental representation subjects build for the interaction.

We distinguish three types of work organization. The first corresponds to the situation where each subject is working on a subset of the problem, i.e. a subset of the intersections in the problem (see figure 1). In this case subjects work in parallel and might have to handle "negative interferences" (Hoc, 1996), the actions of one subject having a negative effect on the successful accomplishment of the other's work. Interaction meters reflect this type of organization by showing equal participation in problem solving actions as well as short phases of talk interrupting the flow of implementation. With the second type of organization the subjects adopt roles: one performs most of the manipulation of the intersections while the other watches and comments. Interaction meters show inequality in the problem solving actions and the talk might be simultaneous to problem solving actions (one subject comments on the other's actions). Finally, subjects may choose not to specialize into a particular role or on a particular subset of the problem and work equally on all subparts of the problem. In this case, interaction meters show equal participation in the problem solving actions and phases of talk. We will refer to these types as 'task', 'role' and 'concurrent' organization respectively.

In order to explore whether the different work organizations we described actually appear in the data that was collected, we created a visualization of division of work. The visualization relies on the number of tuning actions done by each subject on the intersections. We categorized the pairs by visually comparing graphs with each other and assigning them to one of the three categories.

The graphs in figure 4 show prototypical examples of the three types of work division. Figure 4(a) shows a pair that split the task with respect to the intersections. The first subject exclusively made changes to intersections A and B while the other subject exclusively made changes to intersections C and D. Figure 4(b) shows a pair that adopted roles. The first subject did almost all the implementation on all the intersections. Figure 4(c) shows a pair where each subject equally participates to the implementation of all intersections.

Figure 4: Visualization of work division. Circles represent subjects; Rectangles represent intersections (intersections A, B, C and D from left to right). The thickness of lines connecting subjects and intersections represent the proportion of tuning actions performed by the subjects. The proportions sum up to 1 for each intersection. (a) Division by task elements (b) Division by roles (c) No Division, concurrent editing.

(a) Task   (b) Roles   (c) Concurrent
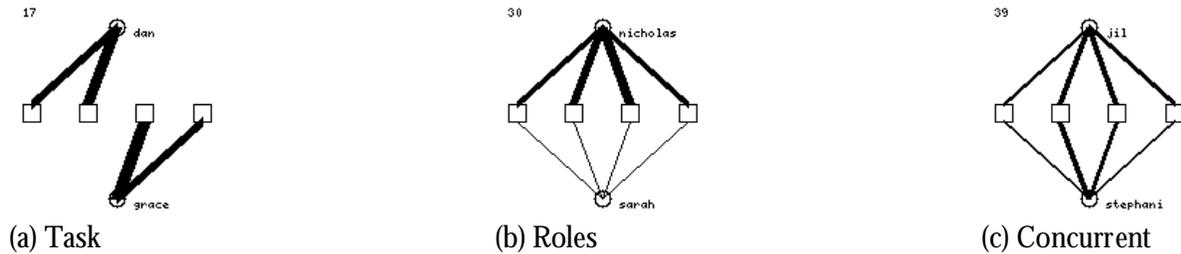
We then decided to formalize the categorization process and give it a quantitative foundation rather than base it on subjective visual classification. The formal definition of the categorization relies on two variables defined in figure 5. The two variables use the difference between the count of tunings made by each subject on each intersection.

Figure 5: Formal definition of work organization. S1 and S2 stand for Subject1 and Subject2. TL stands for Tune Light. $S1TL_1$ represents the number of tuning actions made by Subject1 on intersection 1. S1TL and S2TL stand for the total tunings made by Subject1 and Subject2 regardless of the intersection. The index i (1 to 4, A to D) corresponds to the intersections.

$$\text{SumOfDifferences} = \frac{\sum_i (S1TL_i - S2TL_i)}{S1TL + S2TL} \qquad \text{SumOfAbsoluteDifferences} = \frac{\sum_i |S1TL_i - S2TL_i|}{S1TL + S2TL}$$

The scatter plot in figure 6 shows each pair's position defined by the Absolute Sum Of Differences (ASDiff) (the plain Sum of Differences reflects the domination of a particular subject over the other, which we are not interested in here) and the Sum Of Absolute Differences (SADiff) as x and y coordinates. When ASdiff and SADiff are between 0.0 and 0.5, the subjects participated more or less equally to the implementation. This constraint is

represented in figure 6 by an arc centered at (0;0) and with a radius of 0.5. This corresponds to a situation where either subject at least made 25% of the implementation actions. When ASDiff and SAdiff are equal, pairs are situated on the diagonals drawn on figure 6. This corresponds to one of the subject making more than 75% of the tunings on all intersections. Finally, the pairs that are situated in the center of the graph away from the diagonals correspond to the case where each subject works on separate intersections. These pairs would ideally be situated on the x-axis of the graph at SADiff = 0. They are not exactly situated on the x-axis because one subject usually does more tunings than the other.

Figure 6: Categorization of pairs according to work organization. Each pair is represented by a data point.
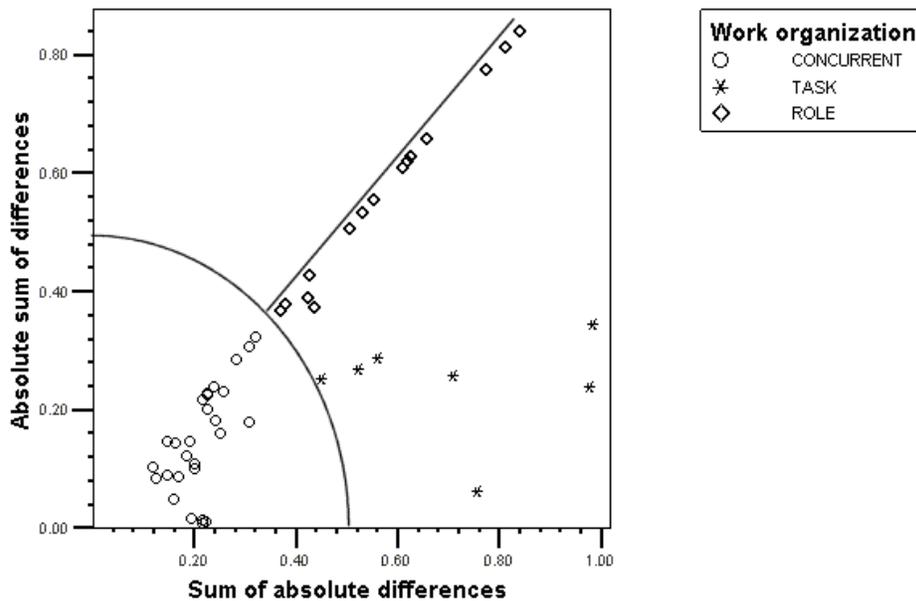
Estimation of participation

The interaction meters were designed in order to help problem solvers build and maintain a model of interaction. Participation is only one aspect of such a model. For instance, a mental model of the interaction might also contain a representation of the roles each subject is playing. The estimation of participation is an indirect measure for the quality of the representation of the interaction held by the subjects that offers the advantage of being easily measurable.

We made a hypothesis that interaction meters give subjects a better estimation of their participation. In order to investigate this hypothesis we compared the observed difference of participation during the interaction with the estimated difference of participation subjects made in the post-experimental questionnaire. We asked subjects to rate four propositions on a 7-point lickert scale ranging from totally agree to totally disagree. The questions were 1) "I made more tunings than my partner"; 2) "My partner made more tunings than me"; 3) "I talked more than my partner" and 4) "My partner talked more than me". We inverted scales 2) and 4) and computed the difference between estimations of self and other. The formulation of these questions was confusing for some subjects, especially when trying to express the fact that both participated equally. As a matter of fact, this would require answering 'I totally disagree' to both questions. Some subjects answered in an inconsistent way. For the analyses presented hereafter we kept only individuals for whom the total of question 1 and 2 (or questions 3 and 4) are between 7 and 9. This constraint expresses the fact that if a subject evaluates himself as having participated more, she also evaluates the partner's as having participated less.

Table 4 shows the correlations between the estimated difference of participation and the real difference of participation for both problem solving actions and dialogue.

Concerning the problem solving actions, it appears that the comparative condition led to the strongest correlation, indicating that subjects estimate the difference in the correct direction (if S1 talked more than S2, S1 makes a compatible estimation). The condition without interaction meter leads to the lowest correlation. This corresponds to our hypothesis. A possible explanation for this result is that subjects are paying more attention to the difference in participation when the interaction meters present a comparison. We also see that successful pairs ('normal' and 'super') make more accurate estimations of their participation than unsuccessful pairs ('failed'). Finally, the pairs who organized their work by adopting roles produce the best estimation of participation. This is not surprising as this organization precisely consists in having one subject do all the tunings.

Concerning the estimation of participation in dialogue, results are less clear maybe because subjects find less difference between themselves and their partner for discussion than they find differences concerning problem solving actions (paired sample t-test d=.94, sd=2.97, p=.004). Also, discussion, by essence, requires both individuals to participate, and estimating a difference in participation for such a collaborative activity might sound misleading to subjects.

Table 4: Correlations between estimated and real difference of participation

		N	Problem solving actions		Dialogue	
			R	P	R	p
Condition	Comparative	19	.785	.000	.582	.009
	Cumulated	21	.601	.004	.208	.366
	Control	23	.508	.013	.772	.000
Success levels	Failed	28	.388	.041	.571	.001
	Normal	14	.766	.001	.575	.032
	Super	21	.792	.000	.669	.001
Work organization	Concurrent	35	.225	.194	.659	.000
	Role	20	.798	.000	.625	.003
	Task	8	.138	.745	-.428	.291

Interaction meters do not foster the production of sequences where task and interaction regulation are integrated

Using the coding scheme presented in the method section, we defined two indices that reflect integration of task and interaction regulation. The first co-occurrence index (cooc1) is obtained by dividing the number of sequences that contain at least one message coded as "Plan" and one message coded as "Organize" by the total duration of the interaction. The second co-occurrence index (cooc2) uses the number of times subjects updated the simulation with new values instead of the duration. This index better reflects the overall pace of planning.

We cannot attribute the increased integration of task and interaction regulation to the presence of interaction meters. Neither cooc1 (F=1.243, p=.299) nor cooc2 (F=.764, p=.472) differ according to the experimental conditions. This is somewhat disappointing because it indicates that the interaction meters did not have the fostering effect on regulation that we expected. Though, we do not definitely reject the potential effect of interaction meters on the type of regulations produced by the pair. There are other factors that might contrast this result and act as enablers for the effect of the meters. Work organization for instance might make the participation meters more relevant to certain pairs. Another hypothesis is that the interpretation of interaction meters in terms of indicators useful to regulate the interaction is too difficult for the subjects. Interaction meters displaying normative information that is directly relevant to the regulation of the interaction might be more effective.

Characteristics of the interaction and levels of success

The results we present now describe the characteristics of interaction that are related to successful completion of the task. Some of these variables could be useful in designing new interaction meters intended to help subjects shape the interaction in a productive way.

Successful pairs talk more than they tune

Overall, it appears that talk helps to succeed. We computed a simple proportion, dividing the number of words by the number of tuning actions. The amount of talk compared to the amount of tuning is larger in successful pairs than in unsuccessful pairs (F=6.137, p=0.05 with LSD post hoc test; 'super' > 'normal', p=.020 and 'super'> 'failed',

p=.001). In the 'super' group, there is more than twice as much words than tuning actions. In unsuccessful pairs, this proportion of talk is inferior to one, meaning that tuning actions were slightly superior to the number of words. The mere frequency of words over interaction time is also related to success. (F=4.128 p=.023 with LSD post hoc test; 'super'>'failed', p=.07 and 'super' > 'normal', p=.055)

Successful pairs produce more detailed plans
We computed two indices reflecting the degree of elaboration of plans.

The first index sums the frequencies of utterances coded as 'plan' that contain at least two of the three possible parameters ('target', 'parameter' and 'value', see Coding section). It appears that there is a tendency for 'super' successful pairs produce more elaborated plans than unsuccessful pairs (F=2.915, p=.066 with LSD post hoc test; 'super' > 'failed', p=.021).

The second index relies on the number of times the subjects updated the simulation. This is an interesting alternative to frequencies based on time because it reflects whether subjects do some planning before updating the simulation. This index is the sum of the division of the number of messages coded as 'plan' that contain at leas two of the three possible parameters by the number of updates. Again, the 'super' group produces more detailed plans than unsuccessful group (F=3.451, p=.041 with LSD post hoc test, 'super' > 'failed', p=.013).

Successful pairs monitor results longer
It appears that successful pairs spend more time looking at the performance, i.e. they look at the system evolve for longer than unsuccessful pairs. (F=4.123, p=.02 with Tamhane post hoc test for unequal variances; 'normal'>'failed', p=.004).

Successful pairs produce more sequences where task and interaction are integrated
To test our main hypothesis, namely that pairs integrating organization and task regulation are more effective we use the co-occurrence indices presented above, namely the number of utterances coded as 'plan' and 'organize' divided by the duration of the interaction (cooc1) and the number of utterances coded as 'plan' and 'organize' divided by the number of times the simulation was updated (cooc2).

It appears that 'super' successful pairs produce this type of planning sequences more often than 'normal' achievers and more often than unsuccessful pairs. (cooc1: F=4.233 p=.022 with LSD post hoc test; 'super'>'failed' p=.01 'super'>'normal' p=.015). 'normal' achievers do not produce integrated planning sequences more frequently than unsuccessful pairs. For the second index (cooc2) the same pattern was found (F=5.571, p=.007 with LSD post hoc test, 'super' > 'failed', p=.002 and 'super' > 'normal', p= 0.43). These results are compatible with our hypothesis.

CONCLUSION
In this paper, we presented results from an experimental study investigating two aspects of metacognition at work in collaborative problem solving, namely, task and interaction regulation. The task we used consisted of controlling a traffic simulation by tuning the green-red periods of traffic lights.

We made the hypothesis that interaction meters would help subjects build and maintain a more accurate model of interaction. In addition to a representation of performance, the tool also contained interaction meters in two experimental conditions. Interaction meters display bar charts that represent subjects' participation in discussion and in problem solving actions. Two types of interaction meters were tested. The 'comparative' version of the interaction meter represents subjects' participation side by side. The 'cumulated' interaction meter represents the pair's activity as a single entity by displaying the number of words produced in the dialogue and problem solving actions. The results concerning estimation of participation in problem solving actions were compatible with our hypothesis, suggesting that the 'comparative' version of the interaction meters is more helpful than the 'cumulated' version and than the absence of interaction meter. However, results for the estimation of participation in discussion were less clear. We focused the discussion of results on subject's estimation of participation in the interaction, but other components of a psychological model of interaction would be interesting to investigate. For instance, we might investigate whether subjects are able to perceive roles and what kind of tool would be useful to raise their awareness about their function in the group. We were not able to show that interaction meters foster the integration of task and interaction regulation. We think that the visualization of participation is not directly usable by subjects to regulate interaction and that it is not directly related to success either. In a future experiment we will use a normative interaction meter that displays a variable directly usable for organizing the interaction. For example, the ratio of problem-solving actions by discussion related actions. It should be straightforward for subjects to deduce from the interaction meter that "we have to talk more" or "we have to try something new" from such a visualization.

Our second hypothesis stated that the co-occurrence of task regulation and interaction regulation would lead to a better performance because it reduces coordination problems. Pairs that solved the problem in less than half the time allocated ('super' achievers) are differing from others by several simple traits: they talked relatively more than they executed problem solving actions; they talked more frequently than other pairs; they produced elaborated plans more frequently. But most important, and supporting our hypothesis, 'super' successful pairs more frequently produced planning sequences containing explicit references to one member of the group. In other terms, when deciding "what to do", these pairs also tend to decide "who does what".

ACKNOWLEDGMENTS

REFERENCES

Amalberti, R., & Hoc, J.M. (1998). Analyse des activités cognitives en situation dynamique : pour quels buts? comment? Le Travail Humain, 61, 209-234.

Artzt, A. F. & Armour-Thomas, E. (1992). Development of a Cognitive-Metacognitive Framework for Protocol Analysis of Mathematical Problem Solving in Small Groups. Cognition and Instruction. 9 (2), 137-175.

Baker, M.J. (1999). Argumentation and constructive interaction. In J. Andriessen & P. Coier (Eds.) Foundations of Argumentative Text Processing. Amsterdam: University of Amsterdam Press.

Barros, B. & Verdejo, M.F. (2000). Analysing student interaction processes in order to improve collaboration. The DEGREE approach. International Journal of Artificial Intelligence in Education, 11, 221-241.

Brown, A. (1987). Metacognition, Executive Control, Self-Regulation, and Other More Mysterious Mechanisms. In F.E. Weinert & R.H. Kluwe. Metacognition, motivation and understanding. Hillsdale, New Jersey: Lawrence Erlbaum Publisher.

Dillenbourg, P., Baker, M., Blaye, A. & OMalley, C. (1995). The evolution of research on collaborative learning. In E. Spada & P. Reiman (Eds) Learning in Humans and Machines.

Grusenmeyer, C. & Trognon, A. (1997). Les mécanismes coopératifs en jeu dans les communications de travail : un cadre méthodologique. Le Travail Himain, 60 (1).

Hoc , J.M. (1996). Supervision et contrôle de processus. La cognition en situation dynamique. Grenoble: PUG.

Hoc, J.M., & Amalberti, R. (1999). Analyse des activités cognitives en situation dynamique: dun cadre théorique à une méthode. Le Travail Humain, 62, 97-130.

Jermann, P., Soller, A., & Muehlenbrock, M. (2001). From mirroring to guiding: A review of state of the art technology for supporting collaborative learning. Proceedings of the First European Conference on Computer-Supported Collaborative Learning, Maastricht, The Netherlands, 324-331.

Nickerson, R.S. (1993). On the distribution of cognition: some reflections. In G.Salomon (Ed) Distributed Cognition (pp. 229-261). Cambridge University Press, New York.

Salomon, G. (Ed)(1993). Distributed Cognition. Cambridge University Press, New York.

Schoenfeld, A. H. (1987). What's All the Fuss About Metacognition ? In Cognitive Science and Mathematics Education. Ed: Alan H. Schoenfeld. London: Lawrence Erlbaum Associates.

Traum, D.R. (1994). A computational theory of grounding in natural language conversation. PhD Thesis & Technical Report 545. University of Rochester. Department of Computer Science.

Trognon, A. (1999). Eléments d'analyse interlocutoire. In M. Gilly, J.-P. Roux & A.Trognon (Eds.), Apprendre dans l'interaction: analyse des médiations sémiotiques. Nancy: PUN.

Figure 7: Snapshot of the simulation interface