MIKE
An Automatic Commentary System for Soccer

Kumiko TANAKA-Ishii Itsuki NODA Ian FRANK
Hideyuki NAKASHIMA Koiti HASIDA Hitoshi MATSUBARA

Electrotechnical Laboratory
1-1-4 Umezono, Tsukuba, Ibaraki 305, Japan.

Abstract

This paper describes MIKE, an automatic commentary system for the game of Soccer. Since Soccer is played by teams, describing the course of a game calls for reasoning about multi-agent interactions. MIKE itself is also a multi-agent system, using a role-sharing structure to mediate between six concurrently running soccer analysis agents, including two agents that carry out high-level expert analysis. MIKE's multi-agent nature mirrors the way that human commentators often perform in teams, and reflects the dynamic nature of the Soccer domain; events may occur at any point of the field at any time, making it difficult to fix viewpoints. We describe MIKE's agents and also discuss how to control the interaction between these agents so that an explanation of a game emerges reactively from the system. We present and evaluate examples of the match commentaries produced by MIKE in English, Japanese and French.

1 Introduction

We describe MIKE, an automatic text commentary system capable of producing output in English, Japanese, and French. MIKE is designed for the specific task of Soccer commentary, and within this domain tackles the problem of explaining and classifying interactions between multiple agents (the name MIKE is an acronym for 'Multi-agent Interactions Knowledgeably Explained'.) In general, the development of multi-agent systems involves research issues such as cooperation protocols, distributed control and effective communication. We chose Soccer as our test domain because it provides a dynamic, real-time environment in which it is still relatively easy for such tasks to be classified, monitored and assessed.

Another advantage of using Soccer as a domain for MIKE is that the RoboCup project [KAK+97] has recently established a common platform for research on the domain. In particular, the logs of games from the 'Soccer Server' [NMHF97, Soc] that controls the simulation (software) league of RoboCup, are freely available and provide information on all objects on the pitch at all points of the game. For this reason, we focused our current project on producing an automatic commentary for the Soccer Server.

Further, note that it is not typically possible to fix just one point of interest in a Soccer game at any one time. For example, real TV coverage of Soccer often employs an announcer to follow the ball-by-ball play, an 'expert' to analyze tactics and team strategy, and a 'spotter' to bring interesting events to the attention of the other two. This team works together in real-time to produce the best possible description of the game they are watching. Thus, Soccer commentary itself is potentially a multi-agent activity, with a variety of points of view needing to be considered in parallel. MIKE takes such a multi-agent approach to describing a game, using a role-sharing structure to mediate between six concurrently running Soccer analysis agents, including two agents that carry out high-level expert analysis. How to reactively select one topic, organize the explanation, and then continue to the next topic is one of the key discussions of this paper.

We proceed as follows. First of all, §2 gives an overall view of our system design. §3 then discusses the agents that MIKE uses to produce basic and expert analysis of Soccer. §4 describes how MIKE combines the output of these agents to produce a coherent presentation, and §5 presents and evaluates examples of the commentary produced by this control structure. Finally, §6 considers other Soccer analysis systems, and §7 draws conclusions.

2 Overall System Design

As we mentioned above, MIKE is designed to produce simultaneous commentary for the RoboCup Soccer Server, originally proposed as a standard evaluation method for multi-agent systems [NM96, NMHF97]. The Soccer Server provides a real-time game log of a very high quality, sending information on the positions of the players and the ball to a monitoring program every 100msec. Specifically, this information consists of:

- **player** location and orientation (for all players),

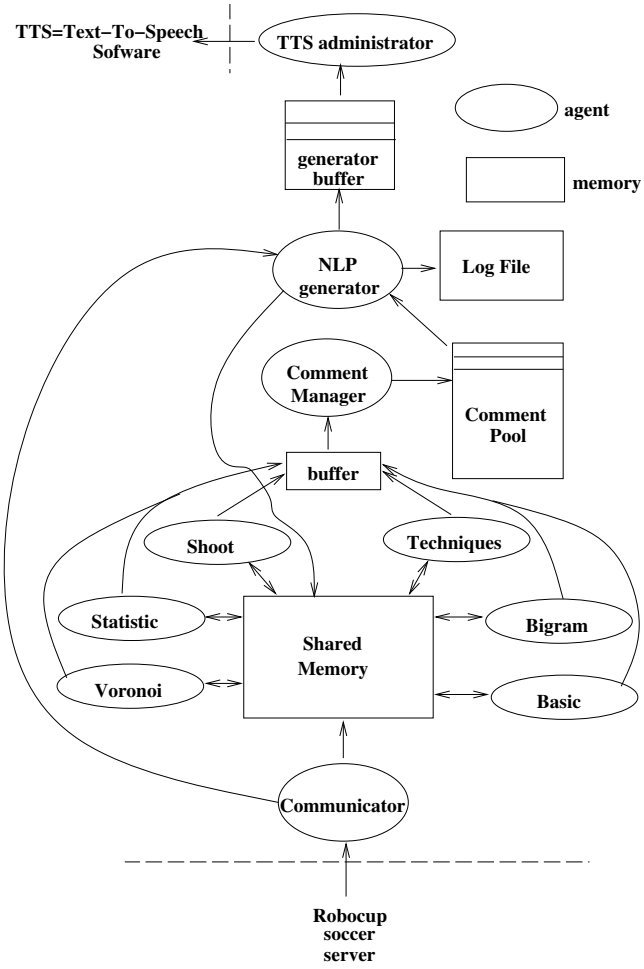

Figure 1: MIKE's structure

- **ball** location, and
- **game** score and play modes (such as throw-ins, goal kicks, *etc*).

This information is clearly more than would typically be available when following a real-world Soccer game, for example by watching TV coverage. However, we expect that even with less information on the course of the game, many of MIKE's commentary techniques will still applicable.

MIKE's structure — a role-sharing multi-agent system — is shown in Figure 1. Here, the ovals represent agents and the rectangles represent data. In total, there are 10 agents, which run concurrently.

In natural language processing, the multi-agent approach dates back to Hearsay-II [EHLR80], which was the first to use the blackboard architecture. The core organization of MIKE, however, is more akin to a subsumption architecture [Bro91], because the agents are regarded as behavior modules which are both directly connected to the external environment (through sensor readings from the shared memory) and can directly produce system behavior (by suggesting commentary). However, MIKE does not exactly fit the subsumption architecture model because the agents can also communicate with each other: there are some portions of the shared memory that are global and some that are exported to only a limited number of agents. This division of shared memory leads to more possibilities for inter-agent communication.

All agent communication is effected via the handling of the internal representation of commentary fragments, which we call *comments*. Comments are represented with a tag and some attributes. For example, a kick by player No.5 is represented as `(Kick 5)`, where `Kick` is the tag and `5` is the attribute. So far, MIKE has 40 sorts of comments, categorized in two ways: as being either of local or global significance, and as being either state-based or event-based (this latter distinction is made more clearly in the following section). Table 1 shows some examples of categorized comment tags.

The actions of the agents in the MIKE architecture can be summarized as follows:

- **Communicator**. Receives log information from the Soccer Server every 100msec and writes it into shared memory. Sends the latest field situation to the NLP generator for reactive topic focusing.
- **Soccer Analyzers**. There are six Soccer Analyzer agents, of which three analyze basic events (shown in the figure as the 'Basic', 'Techniques', and 'Shoot' agents), and three carry out more high-level analysis (shown

Table 1: Examples of how comment tags are categorized

	Local	Global
Event	`Kick`, `Pass`, `Dribble`	`ChangeForm`,
	`ShootPredict`, `PassOnVoronoiLine`	`SideChange`
State	`Mark`, `PlayerPassSuccessRate`	`TeamPassSuccessRate`
	`ProblematicPlayer`	`AveragePassDistance`
		`Score`, `Time`

as the 'Bigram', 'Voronoi', and 'Statistic' agents). The six agents use the information posted to the shared memory by the Communicator, communicate with each other via the shared memory, and also post comments to the Comment Manager (via a buffer).

- **Comment Manager**. This agent receives comments from the Soccer Analyzers and registers them in the Comment Pool.
- **Natural Language Generator**. Selects a comment from the pool that best fits the current state of the game (considering both the situation on the field and the commentary currently being made). Deduction is made from the comment and the other comments in the pool (explained more in 4). The result is then translated into natural language. So far, MIKE just carries out this step with the simple mechanism of template-matching. Several templates are prepared for each comment, and the output can be in English, Japanese or French.
- **Text-To-Speech Administrator**. Synchronizes the Natural Language Generator's output with a text-to-speech software program. Currently, MIKE is only attached Japanese text-to-speech software, Fujitsu FSUNvoice1.0[FSU95].

Since they form the heart of the system, the following section will give a detailed description of the Soccer Analyzer agents. After this, §4 goes on to consider how the inter-agent interactions should be controlled to best produce a coherent explanation of a game from the system.

3 MIKE's Soccer Analyzer Agents

We noted in the Introduction that real Soccer commentary teams often include an announcer to follow the ball-by-ball play and an 'expert' to analyze team strategy and tactics. Broadly, the former of these activities can be characterized as 'event-based' since it involves commenting on and reacting to individual incidents. The latter, on the other hand, is more 'state-based', in the sense that it is based on an analysis of accumulated events on the field of play. The Soccer Analyzer agents used by MIKE can be classified in the same manner.

3.1 Event-based Analysis

Events are identified by three of MIKE's six analyzer agents, as follows:

- The 'Basic' agent identifies simple events in the game, the most notable being ordinary kicks. The current implementation of the Soccer Server only allows players to issue commands to `turn`, `dash`, `say` and `kick`, so there are no complications caused by heading or handballs. Even so, a little care has to be taken over the definition of a 'kick' to exclude events such as the ball changing course by simply bouncing off a player. MIKE uses the practical definition that a player kicks the ball when the following three conditions meet; (1) the player is close enough to the ball to kick it, (2) the direction of the ball changes, and (3) the player's mode is kicking.
- The 'Techniques' agent identifies passes, dribbles, interceptions, one-two passes, and through passes. All of these are defined in terms of patterns of successive kicks.
- The 'Shoot' agent details shots and goals. Currently, shots are detected using the simplest possible criteria of a player's distance from the goal, and his kicking angle towards it. Thus, it is sometimes possible that kicks which are actually intended as passes to team-mates nearer the goal will be interpreted as shots. Actual goals are identified via the decisions of the referee module of the Soccer Server. As soon as a shot is identified a comment is posted to the comment manager. A separate comment may then be posted identifying the goal event.

The comments produced by these three agents can of course directly lead to game commentary. Also, they are essential pre-requisites for producing higher-level descriptions of the game. In MIKE, these higher-level descriptions are produced by state-based analysis.

3.2 State-Based Analysis

MIKE has three agents for carrying out state-based analysis: one based on simple statistics, one based on analyzing each players' passing activity, and one based on a team's global formation.

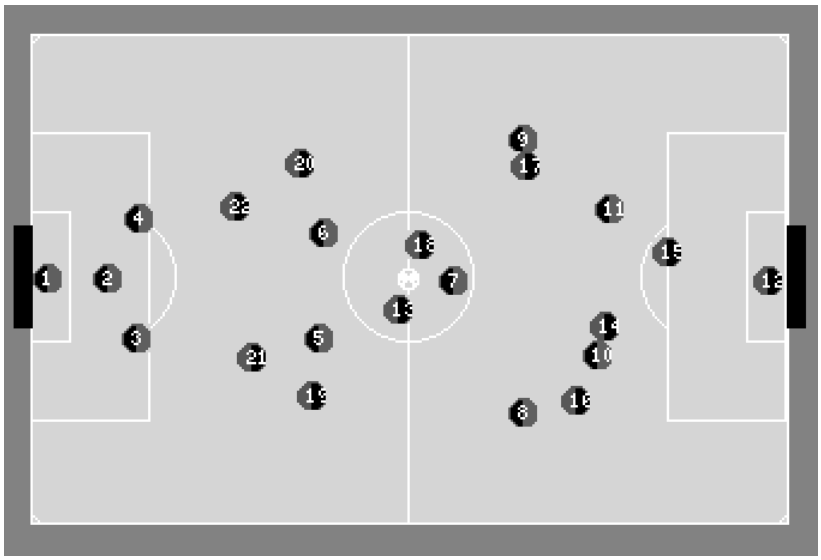

Figure 2: RoboCup'97 final: players' average position

Statistics

MIKE's first state-based analysis agent is the most pedestrian, simply accumulating and analyzing the following statistics during the course of the game:

- each player's average location and its variance. For example, Figure 2 shows the average positions of the players in the first half of the RoboCup'97 World Cup Final held in Nagoya, Japan (players numbered from 1 to 11 play left to right, and those numbered 12 to 22 play right to left). Such analysis is used by MIKE to make simple guesses about the roles being taken by each player.
- the average location of *all* of a team's players taken as a whole (and its variance). This figure indicates whether a team favors defense or attack.
- ball-play average time (for each team). This indicates the speed of a team.
- average pass length.

Analysis based on Ball-play Chains

We regard ball-play chains as first order Markov processes and study the number (probabilities) of transitions. A 24 dimensional matrix (22 players and 2 goals) can be formed and used to describe the activity of each player and to identify successful passwork patterns. Such matrices — which we call *pass bigrams* — are produced automatically in MIKE by the Bigram agent. As an example, consider Table 2, which shows the pass distribution bigram and the pass reception bigram produced by MIKE for part of a real game (again, the first half of the RoboCup'97 World Cup Final).

Only those transitions occurring more than once, plus all goals, are included in a bigram. The left column shows the champion team, Humboldt, whose players are numbered from 1 to 11, and the right column shows Andhill, whose players are numbered 12 to 22. The letter 'G' denotes the scoring of a goal. Transitions involving giving the ball to an opponent are marked with a '+' sign.

MIKE can use this bigram to interpret why Humboldt outperform Andhill. Of course, Humboldt scored seven goals, whereas Andhill scored none, but more than this, MIKE can asses the difference in the passing abilities of the teams by comparing the number of transitions marked with a '+' sign. Also, player No.7 (who can be seen to be a midfielder from the analysis of the Statistics agent in Figure 2), functioned as a playmaker, successfully passing to players No.11 and No.10 (forwards), and even scoring a goal himself. The main goalscorer, Player No.11, scored four times even though Figure 2 suggests he could have been marked by players No.14 and No.15. On the other hand, Andhill did not score any goals although the ball did reach players No.21 and No.22, which Figure 2 identifies as their forwards. Players No.12, 13, 14, 15, 20, 21 often give the ball to the opponents, so they are either severely marked, or just very bad at passing. After full time, this game was eventually won 11-2 by the Humboldt team.

MIKE uses the notion of a *winning passwork pattern* to help interpret a bigram. This is defined as any chain of three players from the same team A, B, C, such that both the transition from A to B and the transition from B to C appear in the bigram, and C scores at least one goal. In the example of Table 2, Humboldt has six winning passwork patterns and Andhill have none, as they failed to score any goals. An example winning passwork pattern is from Player No.6 to No.7 to No.11. MIKE uses winning passwork patterns to predict passwork and shots on goal during a game.

To demonstrate the general applicability of such analysis, we analyzed the logs of all 75 games from the 1997 RoboCup tournament, producing the results of Table 3. As expected, the winning teams have higher numbers of

Table 2: RoboCup'97 final: ball-play transitions

Pass Distribution						Pass Reception					
Humboldt			Andhill			Humboldt			Andhill		
from	to	freq	from	to	freq	to	from	freq	to	from	freq
2	4	2	+12	10	4	4	2	2	+12	10	5
3	5	2	+12	11	3	+4	20	2	+12	11	3
5	8	2	+13	5	4	+5	13	4	+12	8	2
+5	19	2	+13	6	4	+5	21	3	13	19	3
6	7	3	+13	7	3	5	3	2	14	13	2
+6	18	2	13	14	2	+6	13	4	14	15	2
7	8	2	+14	6	2	+6	14	2	14	16	3
7	10	2	+14	7	2	+7	13	3	+14	8	3
7	11	3	+14	10	5	+7	14	2	+15	10	3
+7	18	2	+15	7	2	+7	15	2	+15	11	3
+7	20	2	+15	11	2	+7	17	2	+16	10	3
7	G	1	15	14	2	7	6	3	+18	6	2
8	10	2	16	14	3	8	10	2	+18	7	2
+8	12	2	+17	7	2	8	5	2	+19	5	2
+8	14	3	19	13	3	8	7	2	+20	7	2
8	G	1	+20	4	2	+10	12	4			
10	8	2	+21	5	3	+10	14	5			
+10	12	5				10	7	2			
+10	15	3				10	8	2			
+10	16	3				+11	12	3			
10	G	1				+11	15	2			
+11	12	3				11	7	3			
+11	15	3									
11	G	4									

Table 3: Evaluation of analysis by ball-play chains

	Winning Team	Losing Team
No. of winning passwork patterns	1.13	0.05
No. of players making zero passes	3.20	4.09
Overall pass success rate(%)	46.33	32.34

winning passwork patterns, fewer players that make less no passes to their team-mates, and higher overall pass success rates. MIKE maintains and accumulates these statistics during a game, and uses the results to predict which team will win.

Analysis based on Voronoi Diagrams

Taki [THF96] proposed a model for calculating a player's defensive area based on the player's current speed and orientation. This model generates defense borders in the form of higher ordered curves. In the Soccer Server, however, a player's speed is modeled to decline very quickly. Thus, a player can stop, turn and start running in the other direction with very little notice. Under this condition, we chose a different model to calculate the areas controlled by players on the pitch: Voronoi diagrams.

A Voronoi diagram is a description of a topological structure. Given n points in the plane, which we call *sites*, a Voronoi diagram pre-processes them into a data structure so that given a new query point, q, we can efficiently locate the nearest neighbor of q among the sites. The n sites in fact partition the plane into a collection of n regions, each associated with one of the sites. If region P is associated with site p, then P is the locus of all points in the plane closer to p than to any of the other $n-1$ sites. A graphical rendering of the border of the regions associated with each site is known as the Voronoi diagram for the given sites.

MIKE's Voronoi analyzer agent calculates diagrams for each team, with sites being the players of the team. It does this by applying the technique of [OS95], which has a cost of $O(n)$ on average, with a worst case of $O(n^2)$, where n is the number sites (eleven, for the players of a Soccer team). This technique allows the Voronoi agent to calculate complete diagrams for each team every simulation cycle (100ms). With these, MIKE can determine the defensive areas covered by players and also assess the overall positioning of players. As an example, Figure 3 shows the Voronoi diagram of a certain moment in the first half of RoboCup'97. Humboldt's players are represented by a '+' sign, Andhill's players by a '', and the ball by a ':'

We have already seem how how MIKE can use bigrams identify Humboldt's No.7 as a playmaker. Now we see

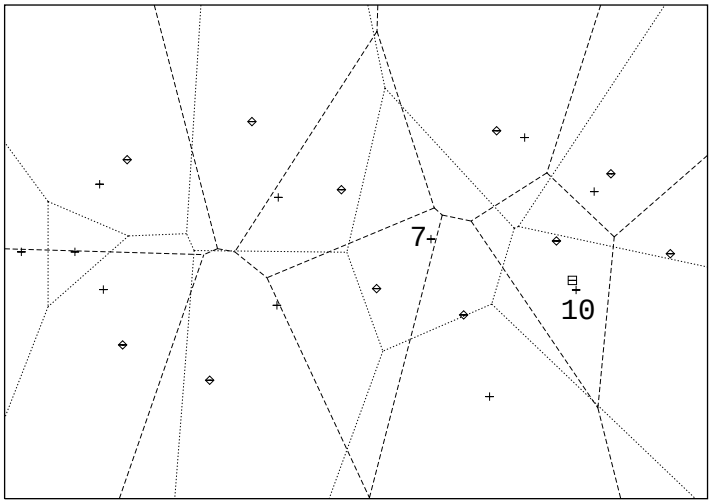

Figure 3: A Voronoi diagram from the RoboCup'97 final

that at the instant depicted by the Voronoi diagram, No.7 is situated nearly on a Voronoi point, as far as possible from the neighboring opponents. Also, No.10 has an excellent chance to shoot at goal: there are no opponents near him and he has the ball. Further, we can see that right in front of the left goal, Humboldt has created a triangular Voronoi area. As the average shape of Voronoi areas is hexagonal, this area demonstrates Humboldt's tight defense. MIKE uses Voronoi diagrams to automatically detect and comment on such features of a game. In particular, MIKE identifies:

- good player positioning: a player is free when he is on the opponent's Voronoi points or lines.
- good passes: defined as those that follow the opponent's Voronoi lines.
- quality of defense: when a player's Voronoi area is too large, the defense is loose; also, a player should mark an opponent in his Voronoi area.

To demonstrate the efficacy of such Voronoi calculations, we again analyzed all the 75 games from the 1997 RoboCup contest, this time counting the number of passes along Voronoi lines. The expectation that winners should make more such passes than losers was confirmed: winning teams average 31.00 passes on Voronoi lines and losing teams average just 22.09 passes. These base values are employed by MIKE for predicting which team is likely to win a game.

3.3 MIKE's Repertoire of Statements

To summarize, MIKE's commentary is produced by combining the analysis of six Soccer Analyzer agents. The repertoire of possible statements that the system can make (40, in total) is outlined below:

- **Explanation of complex events**. Form change, position change, advanced play.
- **Evaluation of team play**. Average form, form at certain moment, players' location, indication of active or problematic players winning passwork patterns, wasteful movements.
- **Suggestions for improving play**. Indication of loose defense areas, suggestions of better locations for inactive players.
- **Prediction**. Game result, shots at goal, pass prediction.
- **Set pieces**. Goal kicks, throw-ins, center kicks, corners, free kicks.
- **Passwork**. Basic tracking of the ball-by-ball play.

4 Overall Control: The Selection of Comments

As we have suggested, coverage of real Soccer games on TV or radio typically involves cooperation between a team of presenters. Crucial to the effective operation of such commentary teams is that the most appropriate commentator is active at each point of the game. For example, the local and event-based comments made by a ball-by-ball announcer are more important when the ball is near to the goal, whereas an expert's more high-level analysis can be presented during lulls in the play or during the natural breaks in the game. Important here is that the audience will prefer explanations to be delivered in coherent wholes, not as the mixture of several contexts at

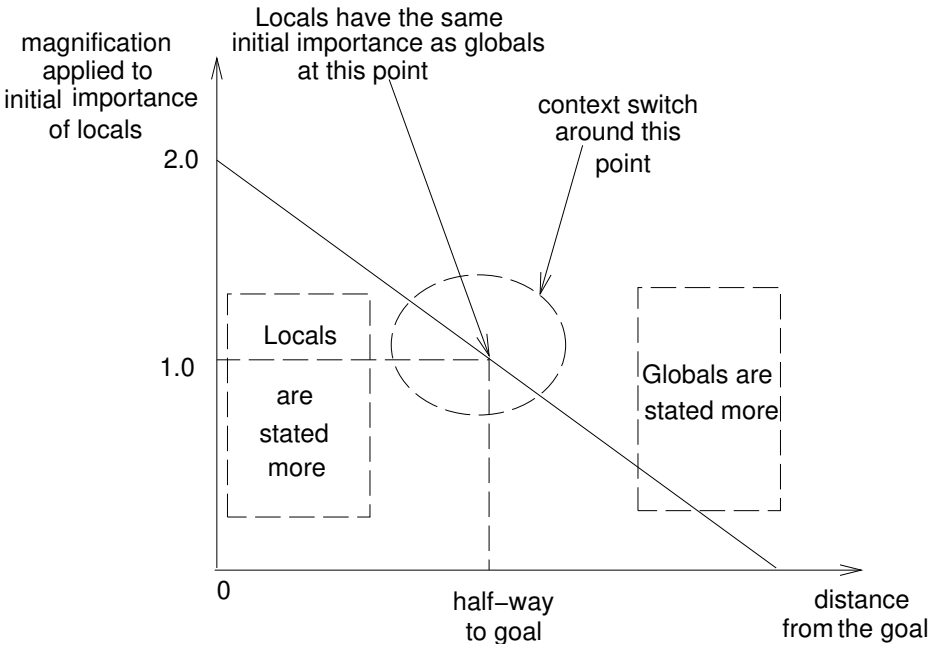

Figure 4: Multiplication of importance values according to the distance from goal

a time. Thus, the members of a commentary team will endeavor not to interrupt the current speaker before he has finished articulating his current train of thought. Further, each member of the team should remember what has already been said, to reduce redundancy and to help indicate what is more important to say next. Of course, some information is always likely to be repeated, since a presentation that contains no redundant information at all tends to be incoherent. Typically, local information may often be repeated, but not usually high-level analysis.

Within MIKE this problem of overall control is dealt with via seven types of controls: an importance value attached to comments, agent cancelation of comments, focusing, continuity control, feedback control, deduction and interruption.

4.1 Importance Values

To establish the relative significance of events, each Soccer Analyzer agent attaches an importance to each of its comments. The Natural Language Generator can then utilize these values to select the comments that are most important to translate into commentary.

The importance of a comment decreases with time while it remains in the Comment Pool waiting to be uttered. When the importance of a comment reaches zero, MIKE deletes it (without ever having stated it). The rate at which importance decreases can be modeled by any (preferably monotonic) function. For simplicity, MIKE's function is currently linear. Since it seems sensible that local comments should lose their importance more quickly than global ones, several functions with differing slopes are used, depending on the degree to which a comment can be considered local or global.

Further, when local comments are initially placed into the Comment Pool a multiplicative factor is applied to their importance value depending on the state of the game (see Figure 4). This factor is designed so that local comments are more important when the ball is near the goal and become less important as the ball moves towards the center line. Global comments are always initialized with the default value.

Some parameters, such as volume, speed, pitch can be selected in current text-to-speech softwares. The importance value is also used to decide these parameters so that the emphasis effect by intonation can be gained.

4.2 Focusing

A heuristic that works well for Soccer is to direct each agent to pay more attention to local events when they occur near the ball. For instance, when the ball is in the left field, it is usually not so helpful for analyzer agents to post comment such as `mark` and `PassSuccessRate` for players on the opposite side of the pitch.

4.3 Continuity

The problem of continuity is addressed by specifying a transitive possibility matrix for the Natural Language Generator. This matrix indicates the desirability of sequentially selecting any two types of comments. For example, when the previous utterance was about the formation of the players, the pass success rate of an individual may

not be an appropriate continuation. Since MIKE has 40 sorts of comments, the possibility matrix is 40×40, with elements that are either 1 (if two comments are regarded as being in the same context) or 0 (for the contrary).

4.4 Feedback Control

Feedback control enables agents to take into account MIKE's past and present utterances when inserting comments into the Comment Pool. The generator broadcasts the current subject to the agents, and they assign higher initial importance values to comments with related subjects. For example, when MIKE is talking about the player No.5, the agents assign a higher importance to comments relating to player No.5.

4.5 Cancelation of Comments

It can happen that when the field situation changes, posted comments become obsolete. In MIKE, as long as a comment remains in the Comment Pool, there is a chance that it will be stated when the system has nothing else to say. Therefore, each agent is given the ability to cancel its previously posted comments. Further, each agent uses MIKE's feedback control to keep a record of stated comments. Referring to this record, the agents can also cancel what should not be repeated.

4.6 Deduction

The former four methods only provide Mike with methods to gather the related facts. These facts often contain the information necessary to analyze the play more deeply. For example, in the discussion of the RoboCup final in §3.2, we saw from the ball-play analysis that player No.7 was Humboldt's chance maker, passing the ball to the three forwards. Also, from the analysis of the Voronoi diagram, we saw that No.7 was situated at the Voronoi point and was free. From these two facts, it is possible to infer that No.7 is a chance maker *because* he was free. Thus, MIKE has the information necessary to suggest that No.7 should be marked by nearby opponents, such as player No.18, to stop Humboldt scoring.

Having c_1 and c_2 as comments, the system has the definition of relationship R among comments i.e. $c_1\ R\ c_2$. We currently have six Rs; reason, prediction, result, adverse, connection and inference. The former five make MIKE to say c_1 and c_2 together, whereas inference transforms $c_1\ R\ c_2 \rightarrow c_3$. Relationship is defined concerning the arguments of comments. The example of player No.7 is expressed as (`Active 7`) R_{reason} (`GlobalPlayerOnVoronoiLine 7`) or (`GlobalPlayerOnVoronoiLine 7`) R_{result} (`Active 7`).

Currently, MIKE only has a primitive deduction engine on a trial basis. An improvement of the deduction mechanism is one of our important future works.

4.7 Interruption

If the play approaches a goal when MIKE is explaining something global, the explanation is interrupted in favor of describing the passwork near the goal.

The mechanism is implemented using the overwrites onto the generator buffer. The data structure of the generator buffer is a ring buffer with a read pointer and a write pointer. Read pointer refers to the very next commentary to be read by the TTS administrator. Write pointer points the end of the former commentary. A commentary, the natural language output transformed from a comment, is cut into the unit of several words and is put in the buffer.

At the usual state, when the ball is not near the goal, the next commentary is put into the generator buffer at the very end of the former statement i.e. at the write pointer. The next commentary is put when the rest of the generator buffer becomes less. When the situation is near the goal, the next statement is overwritten directly from read pointer. The end of the former commentary will be overwritten and disappear without being stated.

5 Evaluation

An example of MIKE's commentary is shown in Table 4. In practice, this output is designed to augment a visual game with a textual description, but since it is impractical to reproduce enough screen-shots to describe the course of the play, we have instead included a column in the table giving some context and further explanations. This particular commentary covers a 30 second period of the same game we analyzed in §3.2: the final of RoboCup'97. Here, the Yellow (left) team is the RoboCup Champion Humboldt, and the Red (right) team is Andhill.

Since this fragment of commentary comes after a reasonable amount of play, MIKE's state-based analyzers have accumulated enough data to recognize winning passwork patterns and predict shots. For instance, when player No.7 gets the ball the system can make a comment about the way that the players No.7, No.10, and No.8 have combined in the past. Also, the pass success rate of player No.3 is stated when he has the ball, demonstrating how local, state-based analyses are chosen when the ball is near the player (or area) of interest. As for global state-based results, these are stated when the ball is in the midfield. (Note that at the beginning of a game, before collecting state-based data, MIKE simply follows the passwork.)

Table 4: Example of MIKE's commentary of 1997 RoboCup final

System output	Further explanation
Goal kick, Yellow-Team,	
Yellow2,	
A pass from Yellow2 to Yellow4,	
Yellow4,	
Interception by the Red-Team,	The Red team quickly makes a poor and ineffectual shot.
Goal kick, Yellow-Team,	
Red11,	The Red team intercepts and shoots.
Keeper Yellow1's defense,	The keeper currently has no 'catch' ability in the Soccer Server
Yellow2,	
Yellow4 receives the pass from Yellow2,	
Yellow4,	
Will Yellow6 make a pass to Yellow7?	It is Yellow6 who has the ball
Red9,	Intercepted by Red9
The Red-Team's play,	
Yellow2,	This time, the Yellow team intercept
Yellow3 usually passes to Yellow5 who gives it to Yellow8, towards the goal,	Prediction of passwork
Yellow3,	
Will Yellow3 make a pass to Yellow5?	
Yellow3's pass success rate is 66 % ,	The ball is in the midfield, so global comments are selected
Yellow7 to Yellow10 to Yellow8 is a promising path to goal for the Yellow team	Yellow7 has the ball
Will Yellow7 make a pass to Yellow11?	Pass prediction
Red1,	Interception near the Red Goal
Red1,	
Yellow11,	Yellow11 gets the ball back
Interception by the Yellow-Team,	
Yellow10 shoots!	Shoot attempt
Red4,	The shot is blocked
Yellow11's shot!	Another attempt
Yellow-Team's 7th score!!	This time, it's a goal
7 to 0!	
Another shot by Yellow11!	
Yellow11 shot straight through the center of the defense!	

あ~、レフトチームのゴールキック、黄色 4 番のキック、おぉっと、黄色 2 番から黄色 4 番へのパス、通りました、赤 11 番、おっ、黄色チームのゴールキック、赤チーム、カットです、黄色 4 番、ああ、黄色 2 番から黄色 4 番へ渡りました、黄色 4 番、黄色チームはサイドに対称な型です、黄色 2 番、黄色 3 番、黄色 5 番、黄色 8 番とパスをまわしてシュートにつなげるのが黄色チームの定石のようです、おっ、黄色 3 番、黄色 7 番から黄色 11 番へパスがよく通ります、黄色 7 番がキック、おっと、赤 1 番は黄色 10 番をマークしています、黄色 11 番のキック、おっ、黄色 10 番のシュート！赤 4 番、あ~、黄色 11 番、シュート！黄色チームの得点です、 7 点です、黄色 11 番、得点、黄色 11 番、センターからシュート、

Figure 5: Japanese output

For comparison, we have included MIKE's French and Japanese descriptions of the same game period in Figure 5 and Figure 6. In general, timing issues resulting from agent concurrency will mean that the content of the commentary generated in any one language will differ slightly from the others, but the bulk of the selected comments always remain the same.

Agent Cooperation

The agent cooperation in MIKE can be visualized by the graphs of Figure 7 and Figure 8, which analyze the comments uttered by the system during the commentary of the first half of the RoboCup '97 final. The graphs are plotted based on the horizontal unit of 2.5 seconds. The vertical axes illustrate the number of comments in each category; in Figure 7 comments are categorized by agents, whereas in Figure 8, they are categorized into local, global and shoot utterances. Note that Figure 7 and Figure 8 both indicate the *selected* comments, not the *posted* comments. The occurrence of each of the seven goals can also be clearly seen by the appearance of the Shoot comments.

Both figures show that MIKE is apt to speak a lot before goals and less after. MIKE is also capable of preserving a context and of grouping utterances of one kind together, since each graph features relatively high and separated peaks, instead of many small and frequent zig-zags. In particular, statistics and global comments are uttered between or just after shots, and the shoot utterances themselves are not disturbed by other comments. These behaviors follow that of actual commentary.

Coup de pied de but, l'équipe jaune, Jaune4, Jaune1, Jaune2, Jaune4 reçoit la balle de Jaune2, Jaune4, L'équipe jaune fait circuler la balle du Jaune6, du Jaune7, le Jaune10 et au goal, Rouge9, L'équipe rouge attaque, Jaune2, Bonne combinaison entre Jaune3 et Jaune5, Jaune3, L'équipe jaune fait circuler la balle du Jaune3, du Jaune5, le Jaune8 et au goal, 66 % de réussite dans les passes pour Jaune3, Jaune7 fait toujours de bonnes passes à Jaune11, L'équipe jaune fait circuler la balle du Jaune7, du Jaune8, le Jaune10 et au goal, Jaune7, Le Rouge1 est pris en charge par le Jaune10, Rouge1, La balle change de camp, Jaune11, Tir de Jaune10! Rouge4, Jaune11, tir! 7 pour l'équipe jaune, Score de 7 à 0, Un autre but de Jaune11! Tir du centre de Jaune11!

Figure 6: French output

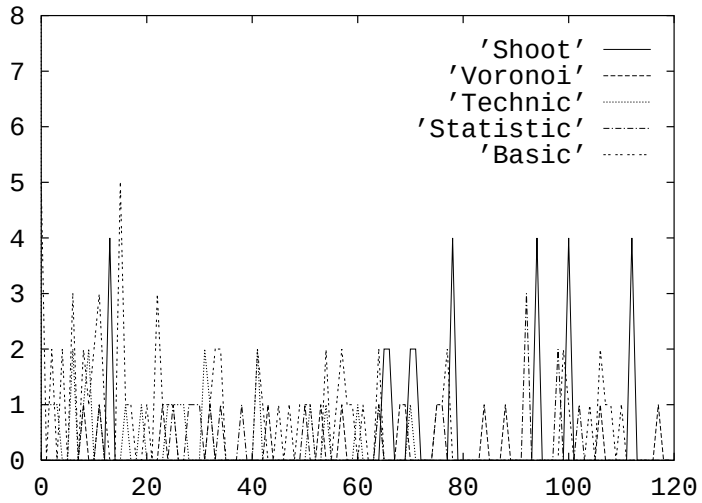

Figure 7: Number of selected comments for each agent (every 2.5 seconds)

6 Related Work

To date, we are aware of only one other attempt at producing an automated Soccer commentary system: the SOCCER system described in [AHR88] (and also more recently in [AHR94]). This program combines a vision system with an intelligent multimedia generation system to provide commentary on 'short sections of video recordings of real soccer games'. The vision system translates TV images into information (actually, a *geometrical scene description* [Neu89]), and the intelligent multimedia generation module then takes this information and presents it by combining media such as text, graphics and video. The VITRA (Visual Translator) [HW94] system, on which SOCCER builds, uses generalized simultaneous scene description to produce concurrent image sequence evaluation and natural language processing. It claims to be the first system that 'automatically generates natural language descriptions for recognized trajectories of objects in a real world image sequence'.

The basic operation of the version of the commentary system described in [AHR94] is event-based. For simultaneous description it has a simple control loop that selects events according to their *topicality*, where topicality is determined by the salience of an event and the time that has passed since its occurrence — a similar notion to the importance values used by MIKE. Unlike MIKE, however, post-game analysis can also be carried out, for which the system has more complicated control structures, plus techniques for searching for events of certain types, modifying the viewpoint from which action is viewed, and enhancing the image to concentrate attention on important features, such as the superimposition of an arrow to indicate the forthcoming movements of ball or players. In terms of simple text, the system produces output such as 'Bommer, the midfield player, passes the ball... to Bosch the outside left... He is attacked by Maller, the outside right.'

Although we are only aware of one automated commentary system for Soccer, systems for automatically producing expert game analysis (rather than just commentary) do exist. Most notably, a commercial software package called 'SECOND LOOK' has been developed by the company SoftSport [Sof]. SoftSport have provided computer analysis services for the American Major League Soccer (MLS), the European Soccer finals in England, and the USA Women's Olympics Soccer Team at the 1996 Olympics Games. SECOND LOOK functions by analyzing game records input manually by a user, from which it produces graphics and charts that analyze the game. The system appears to be based on the basic event of 'kicking the ball', which slightly limits the possible analysis (since the positions of the players who are not kicking the ball at any time are not known) but simplifies the game recording process.

A large proportion of the analysis techniques found in SECOND LOOK were included independently in MIKE. For example, both MIKE and SECOND LOOK compile tables of passes and shots on goal (the bigrams in MIKE) from

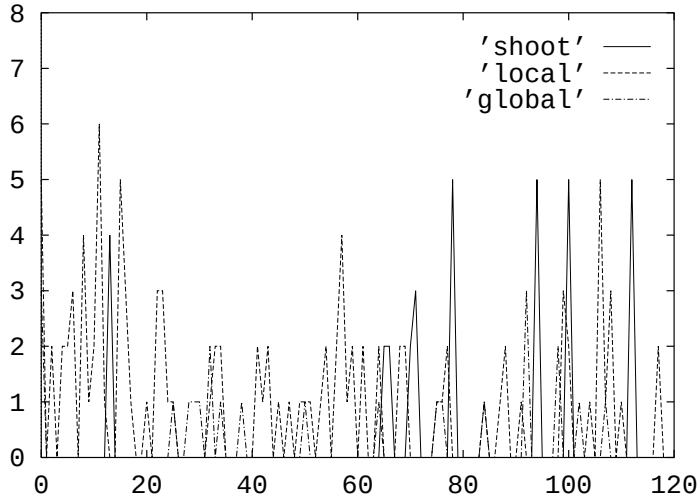

Figure 8: Number of selected local, global and shoot comments (every 2.5 seconds)

which data on completed passes, interceptions and turnover analysis can be retrieved. Both systems also build up comparative databases of the values of such statistics over previous matches (SECOND LOOK using World Cup Soccer matches, and MIKE using matches from the RoboCup). However, SECOND LOOK adds a graphical dimension to its analysis by recording the actual *positions* of all kicks in the game, and compiling graphical displays of, for example, all the passes made by an individual player, or all the shots on goal made by a team. SECOND LOOK also introduces the useful concept of *impact passes* as a further measure of a players' contribution to a game, where impact passes are defined as all those passes made by a team in a series that results in a shot on goal. We intend to add such refinements to future versions of MIKE. We may also be able to make use of formalizations of declarative knowledge about footballing techniques that emerge from the RoboCup initiative itself (see *e.g.*, [Fra97]).

7 Conclusions and Further Work

We presented an automatic commentary system for the RoboCup's Soccer Server with multi-language capabilities. We described the Soccer Analyzer agents that follow the play of a game, and explained how their output was combined within a multi-agent framework to produce a single commentary. MIKE makes ball-by-ball commentary until lulls in the play allow time for giving information on the global play of the teams. In making these global analyses, it brings to bear information from databases of previous Soccer Server matches and also its knowledge of teamwork and Soccer.

Targets for further research on the basic MIKE commentary system include:

- Returning to the former topic after interruption. If the play approaches a goal when MIKE is explaining something global, the explanation is interrupted to describe the passwork near the goal. In the commentary of real Soccer games, the announcers would return to finish such interrupted topics when the interest in the local events had declined. MIKE, however, currently has no mechanism for returning and picking up threads of explanations in this way.
- Adding the ability to incorporate information on any given team's previous performances into a commentary by automatically analyzing logs of their previous games.
- Improving the control structure. For example, Suzuki*et al.*[SIIO97] have proposed new interaction styles to replace conventional goal-oriented dialogue systems. Their multi-agent dialogue system that chats with a human considers topics and goals as being situated within the context of interactions among participants. Their model of context handling uses a subsumptive structure. The solid framework of energy propagation provided by this formalism facilitates context selection between agent interactions that may also be appropriate in MIKE.
- Adding the natural language generation technics such as referential expression, repeats and emphasis.

Note that in addition, portions of the basic MIKE commentary system should also have further applications to other tasks relating to the RoboCup initiative. For example, MIKE's Soccer Analyzer agents can be regarded as general tools for analyzing and evaluating the play of a Soccer Server team. Thus, they could be used to address the tasks of opponent modeling, or even to implement fitness functions for the assessment of teams being produced with evolutionary computation techniques (both opponent modeling and learning are specific challenges laid out in the 1997 RoboCup Synthetic Agent Challenge [KTS$^+$97]).

Finally, the actual purpose of the commentary system could also be extended in more ambitious directions. For example, it could be used to generate short post-match analyses of games (probably incorporating some goal-directed reasoning to achieve goals such as 'describe this headed goal from a corner'). The most ambitious version of such extensions envisages a system combining different presentation modes, such as action replays, written text commentary, diagrams, and speech.

References

[AHR88] E. Andrè, G. Herzog, and T. Rist. On the simultaneous interpretation of real world image sequences and their natural language description: The system soccer. In *Proc. of the 8th ECAI*, pages 449–454, Munich, 1988.

[AHR94] E. Andrè, G. Herzog, and T. Rist. Multimedia presentation of interpreted visual data. In P. McKevitt, editor, *Proceedings of AAAI-94, Workshop on Integration of Natural Language and Vision Processing*, pages 74–82, Seattle, WA, 1994.

[Bro91] R.A. Brooks. New approach to robotics. *Science*, pages 1227–1232, 1991.

[EHLR80] L. D. Erman, F. Hayes-Roth, V. R. Lesser, and D. R. Reddy. The Hearsay-II speech understanding system: Integrating knowledge to resolve uncertainty. *ACM Computing Surveys*, 12(2):213–253, 1980.

[Fra97] I. Frank. Football in recent Times: What we can learn from the newspapers. In *First International Workshop on Robocup in conjunction with RoboCup-97 at IJCAI-97*, pages 75–82, Nagoya, Japan, 1997. Proceedings to be published by Springer-Verlag in the LNCS Series. Also avalaible from the Electrotechnical Laboratory, 1-1-4 Umezono, Japan as Technical Report ETL-97-15.

[FSU95] Fujitsu fsunvoice1.0 japanese speech synthesizer document, 1995.

[HW94] G. Herzog and P. Wazinski. VIsual TRAnslator: Linking perceptions and natural language descriptions. In *Artificial Intelligence Review*, volume 8, pages 175–187, 1994.

[KAK+97] H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, E. Osawa, and H. Matsubara. RoboCup: A challenge problem for AI. *AI Magazine*, pages 73–85, Spring 1997.

[KTS+97] H. Kitano, M. Tambe, P. Stone, M. Veloso, S. Coradeschi, E. Osawa, H. Matsubara, I. Noda, and M. Asada. The RoboCup synthetic agent challenge 97. In *Proceedings of IJCAI-97*, pages 24–29, Nagoya, Japan, 1997.

[Neu89] B. Neumann. Natural language description of time-varying scenes. In D.L. Waltz, editor, *Semantic Structures: Advances in Natural Language Processing*, pages 167–207. Lawrence Erlbaum, 1989. ISBN 0-89859-817-6.

[NM96] I. Noda and H. Matubara. Soccer Server and researches on multi-agent systems. In Hiroaki Kitano, editor, *Proceedings of IROS-96 Workshop on RoboCup*, pages 1–7, Nov. 1996.

[NMHF97] I. Noda, H. Matsubara, K. Hiraki, and I. Frank. Soccer Server: a tool for research on multi-agent systems. Technical Report ETL-97-11, Electrotechnical Laboratory, 1997. To appear in Applied Artificial Intelligence.

[OS95] Y. Oishi and K. Sugihara. Topology-oriented divide-and-conquer algorithm for Voronoi diagrams. *Graphical Models and Image Processing*, 57(4):303–314, July 1995.

[SIIO97] N. Suzuki, S. Inoguchi, K. Ishii, and M. Okada. Chatting with interactive agent. In *Eurospeech'97*, volume 3, pages 2243–2247, 1997.

[Soc] Soccer Server. Full details of the Soccer Server can be found on the Web at `http://ci.etl.go.jp/~noda/soccer/server.html`.

[Sof] SECOND LOOK soccer analysis software. The makers, SoftSport Inc., can be found on the Web at `http://www.softsport.com/`.

[THF96] T. Taki, J. Hasegawa, and T. Furukawa. Development of motion analysis system for quantitative evaluation of teamwork in soccer games. In *Proc. of ICIP'96*, 1996.