To appear in Proc. ACM Mulimedia, Boston, Nov. 1996.

Negotiation for Automated Generation of Temporal Multimedia Presentations

Mukesh Dalal, Steven Feiner, Kathleen McKeown, Shimei Pan, Michelle Zhou
Tobias Höllerer, James Shaw, Yong Feng, Jeanne Fromer
Columbia University
Department of Computer Science
500 W. 120th St., 450 CS Building
New York, NY 10027
+1-212-939-7000
{dalal,feiner,mckeown}@cs.columbia.edu

ABSTRACT
Creating high-quality multimedia presentations requires much skill, time, and effort. This is particularly true when temporal media, such as speech and animation, are involved. We describe the design and implementation of a knowledge-based system that generates customized temporal multimedia presentations. We provide an overview of the system's architecture, and explain how speech, written text, and graphics are generated and coordinated. Our emphasis is on how temporal media are coordinated by the system through a multi-stage negotiation process. In negotiation, media-specific generation components interact with a novel coordination component that solves temporal constraints provided by the generators. We illustrate our work with a set of examples generated by the system in a testbed application intended to update hospital caregivers on the status of patients who have undergone a cardiac bypass operation.

KEYWORDS: media coordination, natural language generation, knowledge-based graphics generation, speech

INTRODUCTION

Multimedia presentations can provide an extremely effective way to communicate information. However, designing high-quality multimedia presentations by hand is a difficult and time-consuming task, even for skilled authors. This is especially true when the presentation involves *temporal media*, such as speech and animation, because the order and duration of actions that occur in different temporal media must be coordinated so that the presentation's goals are communicated coherently. To address the multimedia authoring problem, we are developing a testbed system that uses AI techniques to automatically design temporal multimedia presentations that are customized to a specific situation. Our focus in this paper is on the problem of coordinating how and when information is conveyed by different temporal media to create a coherent and effective presentation.

Our approach is embodied in a testbed system, MAGIC (Multimedia Abstract Generation for Intensive Care) [6], which automatically generates multimedia briefings that describe the postoperative status of a patient undergoing Coronary Artery Bypass Graft (CABG) surgery. MAGIC uses the computerized information infrastructure already present in operating rooms at Columbia Presbyterian Medical Center, which provides a detailed online record of the patient's status before, during, and on completion of surgery. Among the available data are vital signs, administered drugs, intravenous lines, information about devices such as a pacemaker or balloon pump, echocardiogram data, and severity assessments. Our system is designed to tailor briefings for different caregivers, including Intensive Care Unit (ICU) nurses, cardiologists, and residents, all of whom need updates on patient status in the critical hour following surgery.

MAGIC is a distributed system whose components use knowledge-based techniques for planning and generating briefings in written text, speech, and graphics. One of its main contributions is a negotiation process for coordinating the order and duration of actions across different media. Order and duration information are both represented using temporal constraints that are generated dynamically by individual media-specific components. To avoid the expensive replanning that is required when the different components' constraints are mutually inconsistent, each media-specific component in MAGIC produces a prioritized list of partial orderings of its actions. A coordination component negotiates with the media-specific components to arrive at a global partial (or even total) ordering of actions that extends some high-priority partial ordering in each medium. Compatibility among orderings is ensured by explicitly representing the relations between conceptual objects and the media actions that refer to them. Durations are coordinated only after obtaining a compatible ordering of actions.

As described below, our approach is different from most earlier work on coordinating multiple media, in which temporal relations are either defined by hand or exist *a priori* (e.g., as stored video with associated audio). Because MAGIC automatically generates the content and form of all media, temporal relations between media objects are determined only at run-time.

After a discussion of previous work in multimedia coordination, we provide an overview of MAGIC's architecture and introduce a set of examples that will be used to describe the generation and coordination process. We then describe the language-generation and graphics-generation components, and explain how coordination is achieved through negotiation between the generation components and a coordination component.

PREVIOUS WORK

Related work in multimedia systems falls into three broad categories: low level synchronization of stored multimedia, flexible synchronization using hand-specified temporal constraints, and dynamic generation of multimedia. In their unified overview of low-level multimedia synchronization, Steinmetz and Nahrstedt [24] present a four-layer synchronization reference model, an overview of multimedia synchronization approaches, and an account of multimedia presentation requirements; they focus on multimedia synchronization issues for stored multimedia systems. Research on network and operating system synchronization issues, such as feedback techniques and protocols for intermedia synchronization given network jitter (e.g., [18, 19, 20]), also falls within this category. In MAGIC, conceptual objects are generated dynamically, so object duration information and inter-object synchronization points can rarely be stored or computed prior to the generation of a presentation plan.

Although research in the development of multimedia authoring tools also addresses the problem of automatic synchronization, coordination constraints are explicitly stated by the presentation designer and scheduled at the media object level. The media objects could be audio, video segments, or graphics animations. For example, in [12], each media object is associated with a triple: maximum, minimum, and optimum length. The system is able to provide an optimal cost solution that can satisfy all the temporal constraints with fairness in distributing necessary stretching or shrinking across the media objects. Others (e.g., [4], [13]) incorporate unpredictable temporal behaviors in their temporal relation models. Their algorithms can adjust to compensate for the behavior of indeterministic objects at run-time. In contrast, most of MAGIC's temporal coordination and synchronization constraints are dynamically generated, and they are specified at a more detailed level of representation. For example, temporal constraints are specified among words and phrases in speech and among displaying and highlighting in graphics. Media synchronization is controlled by the system at run-time with much finer granularity.

Although [12, 13] allow both qualitative and quantitative temporal constraints, they do not allow disjunctions among those constraints. While the constraint solver in [12], which is based on linear programming, is exponential in the worst case, the solver in [13], which allows flexible quantitative constraints, sometimes provides "false inconsistent" results. In contrast, MAGIC allows disjunctive qualitative constraints using a language that extends Allen's interval algebra [1] by allowing more than two intervals in the same disjunct. MAGIC also allows simple quantitative constraints and uses an efficient but incomplete constraint solver.

Dynamic multimedia generation systems include SAGE [21, 14], COMET [9], and WIP [3], which are knowledge-based multimedia generation systems that coordinate written text with static graphics. Weitzman and Wittenburg [27] also handle media coordination in their work on generating multimedia presentations. They construct a simple network of spatial and temporal constraints for each grammar to accommodate dynamic relationships among presentation elements, but do not support the replanning capabilities we provide in MAGIC. Temporal media introduce additional complexity, and none of these systems support them.

CUBRICON [15] is one of the earliest systems to dynamically generate coordinated speech and graphics. It combines speech with simple deictic gestures and 2D map graphics to guide the user's visual focus of attention. Both speech and graphics are incorporated into a single unified language generator, so there is no need for formal communication between the generation components.

More recently, a system has been developed at University of Pennsylvania that automatically generates conversations between two human-like agents through synchronized speech, facial expressions, and gestures [5, 16, 17]. This system synchronizes 3D graphics with speech by using the timing information generated by the speech generator to decide when and how facial expressions and gestures are produced. For example, if insufficient time is available to generate a gesture, the system may abort the gesture. Since speech controls the actions of the other generators, no negotiation among the generators is required. In contrast, the language and graphics components in MAGIC have equal status, so some mechanism is required to negotiate the order and duration of actions in the presentation.

SYSTEM ARCHITECTURE

As shown in Figure 1, MAGIC consists of a set of components that work concurrently and cooperatively. MAGIC's source of information about a patient's condition is data derived from several online medical databases. Our current *data server* prototype uses a static knowledge source that contains information previously downloaded from these databases and "sanitized" to allow publication. The *data filter* component selects relevant parts of this data and infers new information from it. Using the domain-independent *concept hierarchy* and the domain-dependent *domain hierarchy*, the data filter creates a knowledge representation of the particular patient, which we call the *instance hierarchy*. These hierarchies are all accessible to the other components in MAGIC. The concept hierarchy is a general domain-independent ontology providing a taxonomy for generic information, such as the fact that reals and integers are both numbers (similar to those found in [2, 22]). The domain hierarchy represents domain-dependent medical information using the same classifications used in the clinical information system at Columbia Presbyterian Medical Center.

Based on the user model (e.g., of nurses or cardiologists), *plan library*, and the instance hierarchy, the *general content*

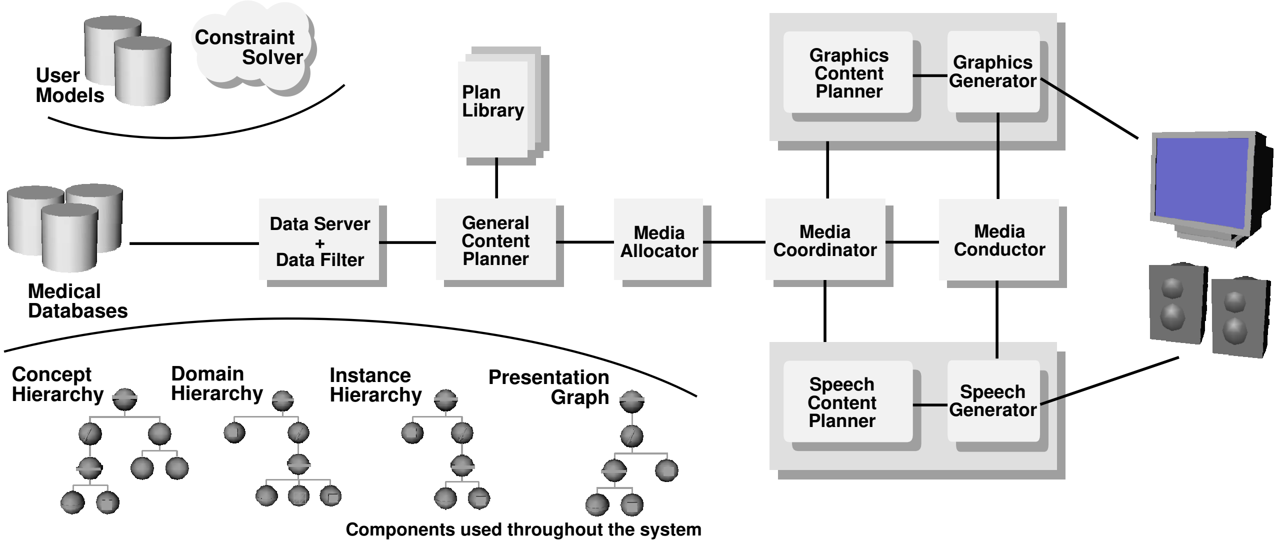


Figure 1: MAGIC system architecture.

planner creates a media-independent plan that expresses the high-level communicative goals to be accomplished by the multimedia briefing. This presentation plan is represented as a directed acyclic graph (the *presentation graph*) and serves as the main data structure for exchanging information among MAGIC's components. The *media allocator* specifies one or more media to express each communicative goal by annotating whether it should appear in one or more media. (Our current implementation uses a simple algorithm based on semantic properties alone.) The full set of annotated communicative goals is then handed over to the media-specific *content planners* and *generators*. A single system-wide *media coordinator* ensures that the media-specific content planners and generators all work out a consistent and synchronized presentation, by allowing media-specific components to access and update the overall presentation plan. Though there are many interesting issues related to the general content planner, the media allocator, and the media-specific content planners, we will not discuss these components in further detail in order to focus on the theme of the paper, temporal coordination.

Several temporal and spatial constraints must be satisfied in the presentation plan. For example, certain tabular information may have to be displayed graphically in a specific spatial order. The media coordinator uses a *constraint solver* component to detect inconsistencies among the different components' plans. The media coordinator handles any inconsistencies by negotiating alternatives with the media-specific content planners. Currently only temporal constraints are explicitly represented, by using qualitative relations based on Allen's interval algebra [1] and simple quantitative information.

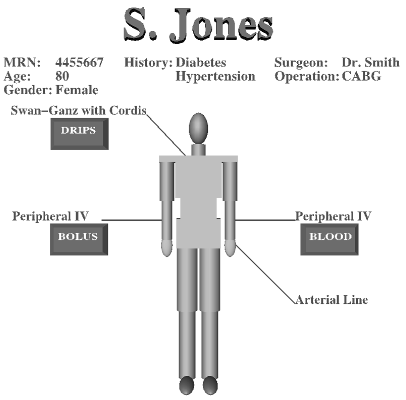
The media-specific content planners and generators engage in fine-grained collaboration to develop a detailed plan for each communicative goal. When the presentation for a goal is agreed upon, it is ready for display. The *media conductor* gives the media-specific generators a ready signal and that part of the presentation is played for the user.

EXAMPLES

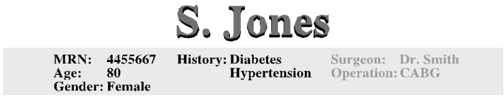
We will rely on two examples, shown in Figures 2 and 3, to illustrate MAGIC's multimedia coordination process. In each example, the goal is to use speech and graphics to communicate information about a patient's demographics, with each spoken reference to visual material synchronized with graphical highlighting. The figures show actual tabular text, speech, and graphics created by MAGIC for the beginnings of two presentations for two different patients. Each figure contains a series of images with the associated speech that is spoken while each is displayed. Two kinds of highlighting are shown: a yellow block that appears around the demographics information to emphasize it, and temporary use of red instead of grey text. (In the black-and-white halftone reproductions included here, the yellow block is rendered as light grey and the highlighted red text is rendered as black.)

The graphics are organized in a standardized spatial layout. Although different data is displayed in both examples, consistent spatial layout of similar information is desirable, since it enables a user to easily find information across multiple patients. Since in each example all the graphics shown are presented before speech starts, synchronized highlighting is used to emphasize parts of the display as they are mentioned in speech.

The decision to use a stylized representation of the patient's body with surrounding information is based on both the information characteristics (e.g., a patient is a physical entity) and the situation/user model (e.g., of the ICU nurses, for whom this particular presentation layout is designed, prefer to see this information arranged relative to the body, but do not need a detailed body model) [31]. Although additional graphics and speech are presented in later parts of both patients' briefings, they are not shown or discussed here. In this paper, we concentrate only on the generation and coordination of speech



(a)



(b) *Speech*: Ms. Jones is an eighty-year-old, diabetic, hypertensive, female patient . . .

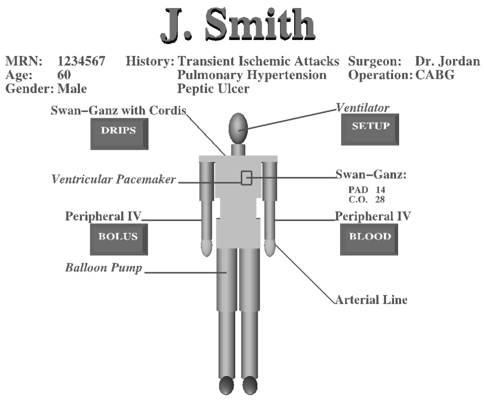
(c) *Speech*: . . . of Dr. Smith undergoing CABG.

Figure 2: Coordinated speech and graphics generated by MAGIC for Jones. (a) Initial display. (b–c) Speech and graphical highlighting for the demographics information at the top of the display.

with the tabular text and graphical highlighting shown at the top of the figures.

Each medium has its own criteria that must be satisfied to produce an ordering of the objects that it presents. The speech components give preference to more concise presentations (e.g., using fewer words). The graphics components give preference to highlighting in a regular, spatially ordered sequence, without jumping around, in a style that highlights in units of an entire column of attribute-value pairs. If speech and graphics were to be planned independently for these examples, speech would generate exactly the sentences shown, and graphics would use a highlighting sequence that preceded from left to right, one column at a time. Although these presentations may be individually satisfactory, when presented together, spoken references and highlighting would differ in order and duration. Thus, to achieve coordination, it is necessary for each medium to constrain the other.

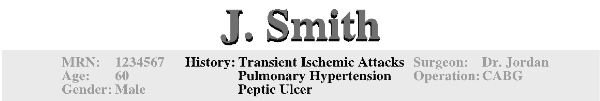
In Figure 2, the ordering of information in speech forces graphics to abandon its preferred ordering and use an alternative instead: the leftmost two attribute-pair columns are highlighted as a block as the words "Ms. Jones is an eighty-year-old, diabetic, hypertensive, female patient" are spoken. The right column is then highlighted as the remainder of the sentence is spoken. No other regular (e.g., left-to-right) highlighting sequence would be coordinated with speech, since



(a)

(b) *Speech*: Mr. Smith is a sixty-year-old male patient . . .

(c) *Speech*: . . . of Dr. Jordan undergoing CABG.



(d) *Speech*: He has a history of transient ischemic attacks, pulmonary hypertension, and peptic ulcer.

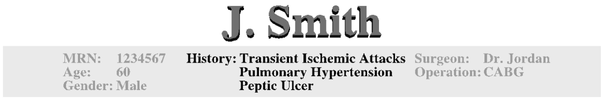
Figure 3: Coordinated speech and graphics generated by MAGIC for Smith. (a) Initial display. (b–d) Speech and graphical highlighting for the demographics information at the top of the display.

the reference to medical history (middle column) is embedded between references to age and gender (left column).

The same left-to-right highlighting would produce an uncoordinated presentation in Figure 3, since the surgeon's name is spoken before the medical history. Instead, graphics must use a less desirable sequence, first highlighting the left column, then the right, and finally, the middle column. (Highlighting the middle and right columns simultaneously is ruled out because the first reference in them is to the right column.) Alternatively, as shown in Figure 4, if the media coordinator requests speech to use a different ordering, speech can find an alternative phrasing that also meets its own constraints in this case.

Given this wording, graphics can use its preferred left-to-right ordering, one column at a time. While in Figure 3, speech constrains graphics choices, in Figure 4, graphics constrains the ordering used by speech, following negotiation. Our work to date gives MAGIC the capacity to generate either of the sequences shown in Figures 3 and 4. (Through user studies we will determine which sequence is more desirable

(a) *Speech*: Mr. Smith is a sixty-year-old male.



(b) *Speech*: He has a history of transient ischemic attacks, pulmonary hypertension, and peptic ulcer.

(c) *Speech*: He is a patient of Dr. Jordan undergoing CABG.

Figure 4: Alternative coordinated speech and graphics generated by MAGIC for Smith. (a–c) Speech and graphical highlighting for the demographics information at the top of the display.

for which users in which circumstances. This will ultimately allow MAGIC to select the appropriate option for the specific situation.)

The temporal constraints used for coordinating speech and graphics are represented using relations in Allen's interval algebra [1]. For example, the qualitative constraint

```
(< name (* age gender))
```

among speech objects indicates that the reference to `name` should be spoken before (<) the references to `age` and `gender`, which in turn may be spoken in any order (`*`). In general, any subset of the thirteen basic relations defined by Allen can be used in place of < and `*`. (We use `*` as an abbreviation for the disjunction consisting of all thirteen basic relations.) We also allow quantitative temporal constraints that specify starting and stopping times of individual intervals in seconds, relative to the start of the entire presentation; for example:

```
(age (start 1.2) (stop 2.7))
```

LANGUAGE GENERATION

Speech is an inherently temporal medium: words and phrases occur in strict temporal sequence, one after the other. The ordering of words and phrases is constrained both by grammatical properties of the language and by communicative goals. The speech components must perform two main tasks to accomplish coordination. First, they must determine possible orderings of spoken references to objects in the accompanying graphics that meet grammatical and communicative constraints. Second, they must determine the duration of each such spoken reference.

These tasks are complicated by the fact that the complete ordering of words is not usually known until all grammatical constraints have been applied and one or more sentences produced. Clearly, negotiating a compatible ordering with graphics at this late stage is undesirable because MAGIC would have to backtrack through the entire process of producing sentences should an incompatibility be detected. Furthermore, given just a string of words, the correspondence is missing between phrases and the objects they reference. To address these problems, we exploit paraphrasing and determination of partial orderings midway through the generation process, by representing and reasoning about possible choices for speech. In this process, MAGIC retains a representation of objects referenced for each phrase, computing the duration of each phrase.

The speech components are capable of realizing the same content in different ways, each of which encodes a different ordering of references. This makes it possible for the media coordinator to select a sentence that matches the ordering requirements from the other media.

Input to the speech components includes objects to be communicated in speech, such as a patient's name, age, gender, medical history, operation and surgeon's name in the patient's demographics section, each of which is represented by a unique identifier. Based on this input, the speech components design several ways to communicate this information. For example, the information given by speech in Figure 2(b–c) can be conveyed in a single sentence or several sentences:

1. Ms. Jones is an eighty-year-old, diabetic, hypertensive, female patient of Dr. Smith undergoing CABG.
2. Ms. Jones is an eighty-year-old female patient of Dr. Smith undergoing CABG. She has a history of diabetes and hypertension.
3. Ms. Jones is an eighty-year-old female. She has a history of diabetes and hypertension. She is a patient of Dr. Smith undergoing CABG.

Additionally, within one sentence, different sentence structures can be produced by changing the word order or paraphrasing. For example, by reordering the words "diabetic" and "hypertensive" in sentence 1 above, we have:

1a. Ms. Jones is an eighty-year-old, hypertensive, diabetic, female patient of Dr. Smith undergoing CABG.

For each sentence shown above, a preference value is produced to indicate how preferable the sentence is from the point of view of the speech components. The speech components are provided with the communicative goal of being concise. Therefore, they prefer one sentence over several, adjectives over prepositional phrases, and prepositional phrases over relative clauses. Since several candidate orders are available to the media coordinator at the same time, this can significantly reduce the number of re-negotiations that are needed when coordination fails.

The object ordering needed for negotiation is produced after planning the sentence and before generating its surface structure. The speech components consist of a speech content planner and a speech generator. The speech content planner determines how much information goes into a sentence. The speech generator includes a lexical chooser that determines overall sentence structure and the words to use, and a sentence generator that uses a grammar to enforce grammatical constraints. The lexical chooser uses unification to generate all possible sentence structures and word choices. The resulting object orders are expressed as partial-order constraints, where the partial orders are used to represent variations that may happen in the final stage of generating the sentence. Thus, each object order may correspond to several sentences that can be generated. For example, when the speech components do not care about the difference in orderings between them, as in sentences 1 and 1a above, they specify the order between "diabetic" and "hypertensive" as `(* diabetes hypertension)`, where `*` indicates that the order of the following objects does not matter.

Since ordering is determined at the point of lexical choice, the speech components have available both the object and the words selected to refer to the object. The speech components keep track of the mapping between the words or phrases in a sentence and their object identifiers. For example, the speech components maintain a representation indicating that "eighty-year-old" represents `age` in the speech input.

The candidate object orders,[1] which correspond to example sentences 1–3 above, produced for the demographics information in the speech output for Figure 2 are:

```
1. (< name age (* diabetes hypertension)
      gender surgeon operation)         [10]
2. (< name age gender surgeon operation
      (* diabetes hypertension))        [5]
3. (< name age gender (* diabetes
      hypertension) surgeon operation)  [4]
```

Note that the speech content planner and lexical chooser determine that `diabetes` and `hypertension` can be referred to using adjectives and thus folded into one sentence in a concise way. Thus, Option 1 is much more preferable than either Option 2 or 3, both of which involve separate sentences and consequently, a number of additional words. Option 1 is given the highest possible weight (10 on a scale of 1–10, shown in square brackets) and Options 2 and 3 are given lower weights, where Option 2 (weight 5) is marginally better than Option 3 (weight 4).

In contrast, for Figure 3, the medical history ("transient ischemic attacks, pulmonary hypertension, and peptic ulcer") cannot be realized in adjectival form; therefore, a separate sentence for the medical history must be generated, which, using the most natural wording, follows references to all information in the left and right columns. Thus, only the last two partial orders above would be generated for Figure 3 (with Smith's medical history items replacing diabetes and hypertension). Again, Option 2 is ranked marginally better than Option 3.

After a compatible total order has been found by the media coordinator, the speech generator generates the surface structure of the set of one or more sentences that meet those constraints. The speech generator then computes duration information for each of the objects referenced in the sentences. The duration, start time, and stop time of each phoneme in the sentences are generated by the speech synthesizer. From this information, the speech generator derives the start and stop time of each word and finally of each object in the given input. As a result, each object identifier referred to in speech is tagged with time information. Note that because durations are provided at the word and phrase level, highlighting can be synchronized to occur for the exact duration of the spoken reference as opposed to the entire sentence, thus yielding a fine level of coordination.

The time information produced by the speech generator for Figure 2, and used by the media coordinator, is:

```
((name 0.5) (age 1.46) (diabetes 2.41)
 (hypertension 3.17) (gender 4.15)
 (surgeon 5.25) (operation 6.29)
 (end-time 7.66))
```

GRAPHICS GENERATION

MAGIC's *graphics content planner* and *graphics generator* handle both time-independent and time-dependent graphics. An example of the former is a static tabular data layout, while examples of the latter include interactive highlighting and more general animation. There is a very high degree of flexibility in how to present several pieces of information graphically in this fashion. Given as input a set of general communicative goals, the challenge is to compute efficiently a course of graphical actions that best fulfills these goals and that at the same time can be smoothly coordinated with the actions performed in other media. To this end the graphics content planner uses a hierarchical-decomposition partial-order planning component [30] that selects visual goals to be accomplished. We have adopted the visual goal categorization developed by Wehrend and Lewis [26]. The planning component computes a partially ordered (and, therefore, flexible) set of graphical actions.

While the speech components order spoken references to objects to obtain possible object orders, the graphics components partially order *graphical actions* that imply possible object orders. Note that the spatial arrangement (*spatial ordering*) of graphical items on the screen is implicitly determined by the graphical actions.

The graphics content planner's input has the same structure as that of the language content planner. For example, when handling the patient's demographics information, the presentation plan advises the graphics content planner to convey information about the patient's age, gender, medical record number (MRN), medical history, operation, and surgeon's name. (Based on domain knowledge provided by our medical experts, the media allocator determines that the MRN is

[1] Although shown here as orderings over objects, speech actually produces orderings over spoken references to objects. For example, instead of `name`, the constraint is over `(refer name)`. This was done to simplify the presentation here.

to be conveyed only by graphics.)

Given the general communicative goal to "emphasize a patient's demographics," the graphics content planner proposes lower-level visual goals. The graphics generator has a set of visual operators that are based on standard graphic design techniques, such as highlighting [31]. The graphics components utilize a set of visual policies to determine what visual operators to use and how to use them to achieve visual goals. In our examples, demographics are displayed in a textual table and a visual policy states that highlighting is appropriate for distinguishing textual table objects. Therefore the *highlight* operator can be selected to accomplish the visual task *distinguish demographics*. Furthermore, the graphics generator uses the action *subhighlight* to further distinguish subparts of the demographics. Thus, a set of actions are proposed to accomplish the current task:

Action1: `(highlight (demographics))`

Action2: `(subhighlight (mrn age gender))`

Action3: `(subhighlight (medhistory))`

Action4: `(subhighlight (surgeon operation))`

The graphics generator communicates with the media coordinator to negotiate the ordering of these actions. A compatible order of the objects involved is produced, utilizing their individual unique object IDs. What is actually communicated to the media coordinator is a set of partially ordered graphical actions. For example, the set of partial orders specified for the actions listed above are:

```
1. (di Action1
       ((< m) Action2 Action3 Action4)) [10]
2. (di Action1
       (* Action2 Action3 Action4))      [7]
```

Here, `di` specifies the relationship *contains* (also known as *during-inverse*), indicating that *Action1* starts before and ends after all the other actions. The relation `m`, called *meet*, indicates that the stop time of *Action2* is the same as the start time of *Action3*, and that the stop time of *Action3* is the same as the start time of *Action4*. The list of operators < and `m` specifies a disjunction. Thus, each action in the list of actions ends either before or at the same time that the next action in the list starts. The first, highly-weighted, partial order highlights information from left to right, while the second indicates equal preference for all other orders.

For Figure 2, after receiving negotiation requests from the media generators, the media coordinator returns to the graphics components a compatible order specified in terms of the objects; in this case:

```
(di demographics
    ((< m) mrn age medhistory gender
           surgeon operation))
```

A complete order of graphical actions is constructed by adapting this compatible object order. If the object order does not agree with the current structure of graphical actions, then the graphics generator needs to fix the current plan. For example, in the order shown above, `medhistory` comes between `age` and `gender`, effectively breaking the structure of *Action2*. In this case, the graphics generator can merge *Action2* and *Action3* together to resolve the conflicts. Thus, a new action is generated to replace *Action2* and *Action3*:

Action5: `(subhighlight (mrn age gender medhistory))`

For Figure 3, the compatible order returned by the media coordinator already agrees with the graphical action structure rated 7 above:

```
(di demographics
    ((< m) mrn age gender surgeon
           operation medhistory))
```

When a satisfactory order has been found, the execution time for each graphical action is estimated within the graphics generator and graphical time constraints are sent to the media coordinator. The time constraints for each graphical action are represented as a time interval specified by its start time and stop time in seconds. Currently, the speech component dictates the time constraints for graphical actions. The graphics generator gets the time constraints from the media coordinator, and then assigns these constraints to each graphical action.

Once a total order of the graphical actions with their time duration constraints has been constructed, the graphics generator sends a ready signal to the media coordinator. Upon the return of a ready signal from the media coordinator, all graphical actions are sent to the rendering component to be realized with the specified timings.

MULTIMEDIA NEGOTIATION

The media coordinator currently coordinates both the order and duration of graphical and speech objects. This is done in two separate negotiation phases: a total ordering of media objects is determined first, and then the object durations are synchronized. Computing durations only after agreeing on a total ordering improves the efficiency of the system because durations (and therefore complete generation) of alternative orderings are not computed at all. Since MAGIC uses automatic generation, when there are incompatibilities in duration, additional material can often be generated to fill the gap.

An important task of the media coordinator is to relate the media objects generated by different media components. For example, the speech generator provides the media coordinator with temporal constraints over spoken references such as "Ms. Jones" and "eighty-year-old." These constraints are represented using actions that specify the underlying conceptual objects, in this case `(refer name)` and `(refer age)`, respectively. Constraints from the graphics generator refer to actions such as `(subhighlight medhistory)`. Note that this becomes complicated because of the hierarchy of objects; for example, in Jones's case, `medhistory` refers to `diabetes` and `hypertension`. The media coordinator correlates the actions provided by speech with the actions provided by graphics, while using subsumption in the hierarchy of conceptual objects.

Because multimedia objects are organized hierarchically and generated automatically, it is important that the constraints provided by the speech generator and the graphics generator be flexible. Regularly backtracking to the planning level because of overly restrictive temporal constraints would be costly in a real-time environment because new object hierarchies may have to be generated. Thus, both the speech and graphics generators provide a list of partial temporal constraints; since they are partial, the temporal constraints can be relaxed if needed. This approach greatly facilitates negotiation between the graphics generator and the speech generator.

For coordinating order, each media generator provides a weighted list of possible partial orders of media actions, ranked according to the generator's preferences. These constraints are expressed in an interval-based model [1] for representing qualitative constraints among temporal intervals. The media coordinator determines a total ordering compatible with the highest-ranked ordering of each medium. If this fails, it negotiates among the speech and graphics preferences until it finds a compatible ordering by systematically traversing the lists. When determined, the compatible ordering is then passed back to the speech generator and the graphics generator.

For Jones's demographics (Figure 2), the highest-ranked constraints for graphics and speech are not compatible. The media coordinator then considers the next ranked graphics constraints, since they have a much higher weight than the next ranked speech constraints. The compatible ordering produced is:

```
(di demographics
    ((< m) name mrn age diabetes
           hypertension gender
           surgeon operation))
```

For Smith's demographics (Figures 3 and 4), recall that the speech constraint corresponding to that ranked highest for Jones is not generated. While Smith's highest-ranked speech and graphics constraints are still not compatible, the differences between the weights of the top two speech constraints and the top two graphics constraints are similar. Therefore, there is not necessarily a clear choice between orders that use the highest-ranked speech constraint and ones that use the highest-ranked graphics constraint. Figure 3 was generated by using the highest-ranked speech constraint (corresponding to Jones's second-ranked speech constraint) and the second-ranked graphics constraint. In contrast, Figure 4 was generated by using the second-ranked speech constraint (corresponding to Jones's third-ranked speech constraint) and the highest-ranked graphics constraint.

After a global total ordering has been agreed upon, each individual media generator computes the duration constraints of its conceptual objects. The media generators provide these duration constraints to the media coordinator, which again uses the constraint solver to determine compatible global start and stop times for each media object. Since graphics does not currently provide any duration constraints, the durations of all objects are computed directly from the speech duration constraints. In particular, the duration constraints generated by speech are used to compute the start and stop times for each of the graphics generator's composite objects. The start time for a composite object c of the graphics generator is computed to be the start time of the first object in the speech generator's object duration information that overlaps with c. The stop time is similarly computed to be the end time of the last object in the speech generator's object duration information that overlaps with c.

In both the Jones and Smith examples, only the speech generator has duration constraints, so the durations of the graphics objects can be computed directly from the speech duration constraints without negotiation. The final start and stop times computed for the speech objects remain the same as specified in the input of the speech generator. The final start and stop times computed for Jones's graphics objects are:

```
((highlight demographics)
 (start 1.46) (stop 7.66))
((subhighlight mrn age gender medhistory)
 (start 1.46) (stop 5.25))
((subhighlight surgeon operation)
 (start 5.25) (stop 7.66))
```

When the media generators are ready to begin the presentation, they each request a ready signal from the media conductor, which waits for both the requests in order to synchronize their start times.

CURRENT IMPLEMENTATION

MAGIC's modules are written in Common Lisp, C, C++, and CLIPS. The entire system is integrated using the ILU (Inter-Language Unification) package [10]. ILU makes it possible for the different components to share data structures and to communicate efficiently across different hardware platforms and implementation languages. The hierarchies and presentation graph described in the section on system architecture are implemented as two multithreaded ILU server modules, one for the three hierarchies and one for the presentation graph. The modules along the main pipeline of MAGIC act as ILU servers as well as ILU clients, so they offer functionality to other modules and also call functions provided by other modules.

The media coordinator currently uses a constraint solver based on *Metric/Allen Time System* (MATS) [11] for solving the temporal constraints and determining a total global ordering. The constraint solver can handle both qualitative constraints for expressing ordering and quantitative constraints for expressing durations. It works by computing the transitive closure over qualitative constraints and then using constraint satisfaction for point-based metric reasoning [29]. The media coordinator, which is implemented in C, runs on a Sun UltraSparc workstation. Currently the general content planner and the media allocator are implemented as one module, using Allegro Common Lisp.

The speech content planner and generator are implemented in Allego Common Lisp and the FUF/Surge package [8]. FUF (Functional Unification Formalism) is a unification-based natural language generator program, consisting of a unifier and a linearizer. Surge is a large, robust unification grammar of English. Both were developed at Columbia University and have been used in a wide range of applications at Columbia and elsewhere. The speech synthesizer is Lucent Bell Laboratories' Text to Speech system. The speech content planner and generator run on a Sun UltraSparc workstation.

The graphics content planner and generator are written in C++ and the CLIPS production system language [23]. We have implemented the knowledge-based design component in CLIPS, and the rendering component using the SGI Open Inventor 3D graphics toolkit [25]. We use a hierarchical decompositional partial order planner [32] that is based in part on the research of [28, 30]. The graphics content planner and generator run on a 250 MHZ R4400 SGI Indigo Maximum Impact.

CONCLUSIONS AND FUTURE WORK

We have described our first steps in the automated generation of multimedia presentations in which multiple media-specific generators negotiate on an equal basis to determine the order and duration of the material presented. This allows MAGIC to automatically determine which media objects should be synchronized and their position, in addition to standard synchronization. This is achieved in an efficient manner through the development of media-specific content planners and generators that can reason about possible orders for presentation and a media coordinator that negotiates between their statements of preferred orderings using temporal reasoning to determine compatibility.

Although the media supported by MAGIC currently include only generated speech, text, and graphics, its negotiation approach can accommodate additional media. In particular, we are working on supporting static images and video to allow the inclusion of material such as x-rays and echocardiograms. We are also interested in making it possible for users to interrupt MAGIC's presentations; for example, to change the topic being presented or how it is presented. Since our negotiation approach does not rely on the specifics of the medical domain, it should extend to other domains as well.

Our constraint solver is currently based on MATS, which is efficient but not complete. That is, there are certain temporal constraints that cannot be expressed (e.g., disjunctions of conjunctions of temporal relations). We are exploring using a new anytime algorithm for reasoning with temporal temporal constraint networks that is based on propositional satisfiability [7]. In addition to our focus thus far on representing and reasoning about temporal constraints, we will also explore using similar approaches to handle spatial constraints on multimedia objects. This will allow MAGIC to reason about the consistency of spatial constraints, the spatial layout of displayed objects, and the interactions among these objects.

Currently, the media coordinator and the media conductor return their results to the speech generator and the graphics generator via ILU. We plan on extending the presentation graph to serve as declarative blackboard representation through which temporal ordering and duration results would be posted. Ultimately, media-specific content planners and generators will also post their constraints, preferences and decisions in the presentation graph so that we have a single representation where different components can inspect the decisions of others.

Finally, we are working on a more sophisticated version of the media conductor, which can handle dynamic synchronization between media. Our work on the presentation graph will help in this task. For example, once duration constraints are represented declaratively in the graph, MAGIC can both interrupt and replay presentations simply by reading the graph. We will also extend negotiation to synchronize durations, if no compatible timings can be determined initially. Furthermore, we will investigate the incorporation of established techniques to alleviate lags that arise from network delays [24].

ACKNOWLEDGMENTS

We are grateful to our collaborators, Profs. Desmond Jordan (Department of Anesthesiology) and Barry Allen (Department of Medical Informatics) at the Columbia College of Physicians and Surgeons, who provided domain expertise in medical and database issues. This research is supported in part by DARPA Contract DAAL01-94-K-0119, the Columbia University Center for Advanced Technology in High Performance Computing and Communications in Healthcare (funded by the New York State Science and Technology Foundation), ONR Contract N00014-94-1-0564, and NSF Grants IRI-94-10117 and GER-90-2406.

REFERENCES

1. J.F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, 1983.

2. J.F. Allen. *Natural Language Understanding*. Benjamin/Cummings, Menlo Park, CA, 1987.

3. E. Andre, W. Finkler, W. Graf, T. Rist, A. Schauder, and W. Wahlster. WIP: The automatic synthesis of multimodal presentations. In M. Maybury, editor, *Intelligent Multimedia Interfaces*, pages 75–93. AAAI Press / The MIT Press, Menlo Park, CA, 1993.

4. M. Buchanan and P. Zellweger. Automatically generating consistent schedules for multimedia documents. *Multimedia Systems*, 1(2):55–67, 1993.

5. J. Cassell, C. Pelachaud, N. Badler, M. Steedman, B. Achorn, T. Becket, B. Douville, S. Provost, and M. Stone. Animated conversation: Rule-based generation of facial expression, gesture and spoken intonation for multiple conversational agents. In *Proc. SIGGRAPH '94*, pages 413–420, 1994.

6. M. Dalal, S. Feiner, K. McKeown, D. Jordan, B. Allen, and Y. alSafadi. Magic: An experimental system for generating multimedia briefings about post-bypass patient status. In *Proceedings American Medical Informatics Association Annual Fall Symposium*, Washington, D.C., October 1996.

7. M. Dalal and Y. Feng. Anytime temporal reasoning based on propositional satisfiability (extended abstract). In E. C. Freuder, editor, *Proceedings of Second International Conference on Principles and Practice of Constraint Programming (CP96)*, pages 535–536, Cambridge, Massachusetts, Aug 1996. Springer.

8. Michael Elhadad. *Using Argumentation to Control Lexical Choice: A Functional Unification Implementation*. PhD thesis, Department of Computer Science, Columbia University, New York, 1993.

9. S. Feiner and K. McKeown. Automating the generation of coordinated multimedia explanations. *IEEE Computer*, 24(10):33–41, October 1991.

10. B. Janssen, D. Severson, and M. Spreitzer. *ILU 1.8 Reference Manual*. Xerox Corporation, Palo Alto, CA, 1993–1995. version 1.8.

11. H. Kautz. *MATS (Metric/Allen Time System) Documentation*. AT&T Bell Laboratories, 1991.

12. M.Y. Kim and J. Song. Multimedia documents with elastic time. In *Proc. ACM Multimedia '95*, pages 143–154, San Francisco, CA, November 1995.

13. N. Layaida and C. Keramane. Maintaining temporal consistency of multimedia documents. In *Electronic Proceedings of the Effective Abstractions in Multimedia Workshop, ACM Multimedia '95*, 1995.

14. V. Mittal, S.F. Roth, J.D. Moore, J.A. Mattis, and G. Carenini. Generating explanatory captions for information graphics. In *Proc. Int. Joint Conf. on Artificial Intelligence*. IJCAI, Montreal, Canada, August 1995.

15. J.G. Neal, C.Y. Thielman, Z. Dobes, S.M. Haller, and S.C. Shapiro. Natural language with integrated deictic and graphic gestures. In *Proc. Speech and Natural Language Workshop*, pages 410–423. Cape Cod, MA, 1989.

16. C. Pelachaud and S. Prevost. Sight and sound: Generating facial expressions and spoken intonation from context. In *Proc. 2nd ESCA/IEEE Workshop on Speech Synthesis*, pages 216–219. New Paltz, NY, 1994.

17. C. Pelachaud and S. Prevost. Coordinating vocal and visual parameters for 3D virtual agents. In *Proc. 2nd Eurographics Workshop on Virtual Environments*. Monte Carlo, 1995.

18. S. Ramanathan and P.V. Rangan. Adaptive feedback techniques for synchronized multimedia retreival over integrated networks. *IEEE/ACM Transactions on Networking*, 1(2):246–260, April 1993.

19. P.V. Rangan and S. Ramanathan. Feedback techniques for continuity and synchronization in multimedia information retrival. *ACM Transactions on Information Systems*, 1993.

20. P.V. Rangan, S. Ramanathan, and T. Kaeppner. Performance of inter-media synchronization in distributed and heterogeneous multimedia systems. *Computer Networks and ISDN Systems*, 1993.

21. S.F. Roth, J.A. Mattis, and X. Mesnard. Graphics and natural language as components of automatic explanation. In J.W. Sullivan and S.W. Tyler, editors, *Intelligent User Interfaces*, pages 207–239. Addison-Wesley, Reading, MA, 1991.

22. S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 1995.

23. Software Technology Branch, Lyndon B. Johnson Space Center. *CLIPS Reference Manual*, June 1993. CLIPS Version 6.0, JSC-25012.

24. R. Steinmetz and K. Nahrstedt. *Multimedia: Computing, Communications, & Applications*. Prentice Hall, 1995.

25. P. Strauss and R. Carey. An object-oriented 3D graphics toolkit. *Computer Graphics (Proc. SIGGRAPH '92)*, 26(2):341–349, July 1992.

26. R. Wehrend and C. Lewis. A problem-oriented classification of visualization techniques. In *Proc. 1st IEEE Conference on Visualization: Visualization '90*, pages 139–143. IEEE, Los Alamitos, CA, October 1990.

27. L. Weitzman and K. Wittenburg. Automatic presentation of multimedia documents using relational grammars. In *Proc. ACM Multimedia '94*, pages 403–412, San Francisco, CA, October 1994.

28. D.E. Wilkins. *Practical Planning: Extending the Classical AI Paradigm*. Morgan Kaufmann, San Mateo, CA, 1988.

29. E. Yampratoom and J.F. Allen. *MATS (Performance of Temporal Reasoning Systems)*. Computer Science Department, University of Rochester, 1993. TRAINS Technical Note 93-1.

30. R.M. Young, M.E. Pollack, and J.D. Moore. Decomposition and causality in partial-order planning. In *2nd Int. Conf. on AI Planning Systems: AIPS-94*, pages 188–193. Chicago, IL, June 1994.

31. M. Zhou and S. Feiner. Data characterization for automatically visualizing heterogeneous information. In *Proc. INFOVIS '96 (IEEE Symp. on Information Visualization*, San Francisco, CA, October 28–29 1996.

32. M. Zhou and S. Feiner. Top-down hierarchical planning of coherent visual discourse. In *Proc. IUI '97 (1997 Int. Conf. on Intelligent User Interfaces)*, Orlando, FL, January 6–9 1997.