From: AAAI-87 Proceedings. Copyright ©1987, AAAI (www.aaai.org). All rights reserved.

Filming a Terrain under Uncertainty Using Temporal and Probabilistic Reasoning

Raymond D. Gumb
Department of Computer Science
University of Lowell
Lowell, Massachusetts

Abstract

We address the problem of interpreting sensor data under uncertainty, using temporal and spatial context to facilitate the identification of objects. We seek to identify the type of an object presented in an ambiguous image by reasoning about conditional probabilities and the possible movements objects can make. A conditional probability (that an object is of a certain type given that some of its properties have been recognized) is used in conflict resolution, and an object is assigned an alternative type when an impossible movement is detected. Think of a map as being a frame and a sequence of frames as being a film. The idea is to construct a consistent and plausible (coherent and highly probable) film in which an object of one type does not mysteriously change into an object of another type.

I. Introduction

In this paper, we describe TEMPRO, a system that employs temporal reasoning and probabilities in conflict resolution. [1] TEMPRO focuses on the information management aspects of interpreting sensor data under uncertainty. TEMPRO uses conditional probabilities to order conflicting rules, and diachronic inconsistencies (impossible movements) to trigger the selection of alternative rules.

TEMPRO has been tested in a Monte Carlo simulation. Sensitivity analysis of the experimental results indicates that the system would be less reliable if checks for diachronic consistency were not in place, and both less reliable and more inefficient if the conditional probabilities were ignored in conflict resolution.

The development of TEMPRO was motivated by concerns similar to those motivating such works as [Ferrante, 1985] and [Durfee and Lesser, 1986]. TEMPRO's error correction facilities bear some similarity to the devices of [Ferrante, 1985], which integrates techniques for reasoning about uncertainty and constraint propagation. However, the constraints embedded within TEMPRO are of a temporal as well as of a spatial nature. In the terminology of [Durfee and Lesser, 1986], we consider only reasoning centralized at a single node.

The logical and probabilistic features of TEMPRO can be formalized in a natural manner. Systems similar to TEMPRO can be applied whenever conditional probabilities can be ordered using, say, the methods of [Nilsson, 1986] for probabilistic logic (called *probabilistic semantics* in some of the literature [Leblanc, 1981]). A necessary condition for the application of systems similar to TEMPRO is that universal laws codifying the rules be expressible in a language having a probabilistic semantics. For example, first-order languages and most of the usual first-order intensional languages, such as the one implicit in this paper, have a probabilistic semantics, but, for second-order languages, no probabilistic semantics is known.

TEMPRO can be formalized in terms of the probabilities of alternative Hintikka model systems in a quantified temporal logic with identity and a past tense operator. In formal terms, we construct the most probable Hintikka model system [Leblanc, 1981] (the most probable corrected film) extending a given consistent evolving theory [Gumb, 1978] (a given noisy film).

II. Objectives

TEMPRO is designed to determine the types of objects situated within a two-dimensional world. The two-dimensional world consists of areas laid out in a grid, with zero or more objects contained within an area. Some of the objects are *permanent* (i.e. have a fixed location), whereas other objects are *mobile*. The *known objects* are permanent objects whose existence has been previously established (i.e. prior to the simulation). Permanent objects which are not known and all mobile objects are called *unknown objects*. An area containing one or more unknown objects is said to be *occupied*.

During one unit of time, a mobile object can move to an (immediately) adjacent area (in a horizontal, vertical, or diagonal direction) or it can remain stationary, depending upon its type. An inspector is assigned the task of filming the terrain. The inspector is restricted to moving in a

---

[1] This research was supported by grant 95-2931 from Sandia National Laboratories.

straight line from one area on the grid to another at the rate of one unit of distance per unit of time. In particular, the inspector travels down row 0 and, at time t, is located in area $(0, t)$. The inspector films the terrain in his field of vision, taking snapshots at the rate of one frame per unit of time. The inspector's field of vision is limited, as each snapshot covers only the area where the inspector is currently located and the adjacent areas. The inspector is given (1) an initial map showing the location of the known objects on the terrain and (2) instructions to traverse a path of length n. On his its path, the inspector's objectives are (1) to film as faithfully as possible both the permanent and movable objects and (2) to construct a more complete map of the permanent objects. The inspector is provided with TEMPRO for use as an error correction system.

In the simulation, the identification of unknown objects takes place in the presence of uncertainty. The type of an object is determined by the properties it has, and an error occurs when information regarding an object's properties is lost in the sensor input, making the object's type indeterminate. If the inspector was located at the previous moment in the same area where the object in question is now, TEMPRO might be able to correct the present frame by reasoning about what objects in the immediately preceding frame could now be in that area. (Note that, in the previous moment, the area where the object is now and every area adjacent to that area were in the inspector's field of vision. Hence, in the preceding frame, the inspector could see every object which now could be in the area in question, and the area in question is included in the present frame.) Similarly, TEMPRO might be able to correct past frames based on the present and future frames.

Sometimes, TEMPRO is unable to identify the type of an unknown object with certainty. However, knowing some of the properties of an object allows TEMPRO to make informed guesses about its probable type. TEMPRO is given conditional probabilities to facilitate its guesses. Even when temporal reasoning can identify the type of an object with certainty, the conditional probabilities are useful because they are used in conflict resolution, minimizing the need for chronological backtracking.

The current goal of TEMPRO is to correct errors where, for any occupied area, at most one of its objects can be in error, and, for any object in error, at most one of its properties cannot be identified.

III. The Four Phases in the Simulation

Phase 1: *History Generation*	**Phase 2:** *Noise Generation*	**Phase 3:** *TEMPRO*	**Phase 4:** *Evaluation*
User Input History Correct Map	Noisy Film	Corrected Film Corrected Map	Grade Peformance Index

Figure 1: The Four Phases in the Simulation

Figure 1 depicts the four phases in the simulation testing TEMPRO's error correction abilities. In phase 1, the user supplies a (possibly incomplete) map of the permanent objects and other information which the system uses to generate the history and a correct film of the terrain. Errors are introduced at random in phase 2, resulting in a noisy film, and, in phase 3, TEMPRO attempts to eliminate these errors, producing a corrected film and a more complete map of the permanent objects. In phase 4, TEMPRO's corrected film is checked for correspondence with the correct film (phase 1) and evaluated with a grade and a performance index. The grade gives the accuracy of TEMPRO's corrected film, and the performance index measures TEMPRO's efficiency.

In the first phase, generating the history of the terrain, the user is asked to supply the following information:

1. the number of time periods (n),
2. the location of the known objects on the terrain,
3. the average number of unknown objects per area,
4. for each type, the absolute a priori probability that one of the unknown objects is of that type, and
5. the absolute probability of losing information regarding an occupied area.

The information entered in step (2) provides the initial map of the permanent objects. The number of unknown objects is determined by the information given in steps (1) and (3), and the total number of objects is the sum of the known and unknown objects. Using the information given in step (4), the system chooses at random the type of each unknown object, and then each object is placed at random on the grid, which, together with the initial map of permanent objects, gives the initial (time 1) configuration of the objects on the terrain. Proceeding inductively, the next configuration (time $2, \ldots, n$) of objects on the terrain is obtained by choosing, for each object, one of its possible moves at random.

In phase 2, generating noise in a frame, attention is restricted to the inspector's field of vision. For each point in time t ($1 \leq t \leq n$), the restriction of the configuration of the objects on the terrain to these areas gives the correct frame at time t. A noisy frame is generated from a correct frame by selecting occupied areas (at random) for an error using the error rate specified in step (5) of phase 1, selecting an unknown object for an error in each selected area, and losing one property of each selected object.

In phase 3, correcting errors using temporal and probabilistic reasoning, TEMPRO's corrective action depends upon the time. First, without using any temporal reasoning, TEMPRO determines all possible object configurations that are compatible with the information provided in the noisy frame. Second, TEMPRO orders the possible configurations on a list using conditional probabilities computed from the information entered by the user in step (4) of phase 1. There is one such list of all possible configurations for each time from 1 to n. Third, TEMPRO removes the configuration first on the list, taking this (for the moment at least) to be the corrected frame. If this is time 1 and the list for time 1 is empty, TEMPRO terminates, reporting an error in its program logic. If this is time 1 and the list for time 1 is not empty, TEMPRO proceeds to time 2. If this is time t, $t > 1$, and the list for time t is empty, TEMPRO backtracks to time $t-1$, the configuration first on the list for time $t-1$ is removed and taken to be the (new) corrected frame. If this is time t, $t > 1$, and the list for time t is not empty, the first configuration on the list is removed and checked for compatibility with the corrected frame for time $t-1$. If it is not compatible, it is rejected, and the next configuration on the list is removed, taken to be the corrected frame, and checked. If it is compatible and $t < n$, TEMPRO proceeds to time $t+1$. If it is compatible and $t = n$, TEMPRO reports a map of the permanent objects on the terrain along his path, and the simulation proceeds to phase 4.

The final corrected film (i.e. the final sequence of corrected frames) and the more complete map of the permanent objects are printed. The more complete map gives the location of the known objects as well as the location of those unknown objects that are judged to be of permanent. The final corrected film represents a consistent and plausible (highly probable if not completely correct) evolving theory [Gumb, 1978]. The user is given the option of tracing the corrected frames as they are selected.

In phase four, the correct and corrected frames are compared, and TEMPRO is assigned a grade and a performance index. The grade is the number of errors in the noisy film that were properly corrected in the corrected film divided by the total number of errors in the noisy film. The performance index is an ordered pair (b, r), where b is the number of backtracks and r is the number of rejected configurations. A rough ranking of the TEMPRO's performance under various conditions can be had by arranging performance indices in lexicographic order. The number of backtracks b is the first item in the ordered pair constituting the performance index because backtracking debilitates efficiency as well as real-time veracity.

Type	Properties	
t_1	p_1	p_3
t_2	p_1	p_4
t_3	p_2	p_3
t_4	p_2	p_4

Figure 2: Types and their Properties

Universal Law	Rule Pair
If $p_1(X)$, then $t_1(X)$ iff not $t_2(X)$.	$<t_1(X)$ if $p_1(X)$; $t_2(X)$ if $p_1(X)>$
If $p_2(X)$, then $t_3(X)$ iff not $t_4(X)$.	$<t_3(X)$ if $p_2(X)$; $t_4(X)$ if $p_2(X)>$
If $p_3(X)$, then $t_1(X)$ iff not $t_3(X)$.	$<t_1(X)$ if $p_3(X)$; $t_3(X)$ if $p_3(X)>$
If $p_4(X)$, then $t_2(X)$ iff not $t_4(X)$.	$<t_2(X)$ if $p_4(X)$; $t_4(X)$ if $p_4(X)>$

Figure 3: TEMPRO's Rules are Extracted from Universal Laws

IV. Types of Objects

To illustrate TEMPRO's error correction techniques, we consider the following simple universe: There are only four types of objects ($t_1,\ldots,t_4$) and four properties ($p_1,\ldots,p_4$), which characterize the four types. In Figure 2, note that each type is characterized by two properties.

If information regarding a property of an object is lost in the sensor input, TEMPRO can narrow the object's possible type down to two types. For example, if an unknown object is really of type t_1 and property p_1 is lost, the inspector can determine that the object is either of type t_1 or t_3 because the inspector knows that property p_3 holds. The information in Figure 2 determines four universal laws (Figure 3) that state that, if one of the four properties hold of an object **o**, then **o** is of exactly one of two types. From each universal law, a pair of conflicting TEMPRO rules is extracted as shown in Figure 3. The universals laws are said to *codify* TEMPRO's rules.

Within each pair of rules, conditional probabilities resolve conflicts. For example, regarding the first pair of rules, if object **o** is observed to have property p_1 and the conditional probability of an object's being of type t_1 given that it has property p_1 is greater than the conditional probability of its being type t_2 given p_1, then object **o** is taken to be of type t_1.

Permanent objects are of type t_1. Objects of types $t_2 - t_4$ are mobile and, during one time period, can remain stationary or move to an adjacent area.

V. The Compatibility Checks for Diachronic Consistency

Suppose k errors occur in a noisy frame, and, if $k \geq 1$, that the i-th object in error ($1 \leq i \leq k$) is observed to have exactly one property. Then, the unknown object can be of one of two possible types. In general, there are 2^k possible configurations of the objects (i.e. possible corrected frames). Each possible configuration is compatible with the information provided in the noisy frame. In a possible configuration (prior to making the compatibility checks), if $p_{i_1}, \ldots, p_{i_k}$ are the properties observed of the k unknown objects in error, $t_{i_1}, \ldots, t_{i_k}$ are their assumed types, and $P(t_{i_j}, p_{i_j})$ is the conditional probability of t_{i_j} given p_{i_j}, then (assuming independence) we have $P(t_{i_1}, p_{i_1}) \times \cdots \times P(t_{i_k}, p_{i_k})$ as the probability of this configuration. The ordering of the possible configurations induced by these probabilities is used in conflict resolution as described earlier.

Each of two frames, when viewed in temporal isolation, might be (synchronically) consistent, but, when viewed in temporal succession, might not be (diachronically) consistent. The possible movements of objects serve to determine compatibility checks for adjacent frames in a film. For example, an unknown mobile object located in area (i, j) has 9 possible movements available to it if it is not on an edge of the grid. The nine areas to which it can move are $(i-1, j-1)$, $(i-1, j)$, $(i-1, j+1)$, $(i, j-1)$, (i, j), $(i, j+1)$, $(i+1, j-1)$, $(i+1, j)$, and $(i+1, j+1)$.

TEMPRO employs three compatibility checks concerning the possible movements of objects in the inspector's field of vision. If the simulation is just beginning and so the time is $t = 1$, then TEMPRO can make no compatibility checks because there is not (yet) a past frame to provide temporal context. If the time is $t \geq 2$, there are three compatibility checks:

1. In each of the 6 areas covered in both the frame for time $t-1$ and and the frame for time t, there must be the same number of permanent (type t_1) objects. (The 6 common areas are $(-1, t-1)$, $(-1, t)$, $(0, t-1)$, $(0, t)$, $(1, t-1)$, and $(1, t)$).
2. For each type from t_2 to t_4, in the frame for time $t-1$, the number of objects of that type located in the area $(0, t)$ must be less than or equal to the sum in the frame for time t of the objects of that type in that and adjacent areas (i.e. the areas $(-1, t-1)$, $(-1, t)$, $(-1, t+1)$, $(0, t-1)$, $(0, t)$, $(0, t+1)$, $(1, t-1)$, $(1, t)$, and $(1, t+1)$).
3. For each type from t_2 to t_4, in the frame for time t, the number of objects of that type located in the area $(0, t-1)$ must be less than or equal to the sum in the frame for time $t-1$ of the objects of that type in that and adjacent areas (i.e. the areas $(-1, t-2)$, $(-1, t-1)$, $(-1, t)$, $(0, t-2)$, $(0, t-1)$, $(0, t)$, $(1, t-2)$, $(1, t-1)$, and $(1, t)$).

time	1			2			3		
row -1	?								
row 0		I		C	I	?		I	
row 1						B		?	T
column	0	1	2	1	2	3	2	3	4

Figure 4: A Noisy Film

time	1			2			3		
row -1	C								
row 0		I		C	I	T		I	
row 1						B		B	T
column	0	1	2	1	2	3	2	3	4

Figure 5: A Corrected Film

If $t > 1$ and the corrected frame for time t is not compatible with the corrected frame for time $t-1$, TEMPRO rejects the corrected frame for time t.

VI. An Informal Illustration of the Compatibility Checks

Consider the noisy film in Figure 4 consisting of the frames for times 1, 2, and 3. In each of the three frames, the inspector (**I**) is in the middle of the areas in his field of view. A question mark (**?**) indicates those areas in which information about an object has been lost. The following objects are observed with certainty: A boulder (**B**) at time 2 in area $(1, 3)$, a car (**C**) at time 2 in area $(0, 1)$, and a truck (**T**) at time 3 in area $(1, 4)$. TEMPRO's compatibility checks enable the types of all three unidentified objects (**?**'s) to be determined:

1. In frame 1, a car (**C**) is in area $(-1, 0)$ because the car at time 2 in area $(0, 1)$ must have come from there (compatibility check (3)).
2. In frame 2, a truck (**T**) is in area $(0, 3)$ because it must have moved at time 3 to area $(1, 4)$ and, at time 3, a truck is the only object in area $(1, 4)$ (compatibility check (2)).
3. In frame 3, a boulder (**B**) is in area $(1, 3)$ because it was there at time 2 (compatibility check (1)).

TEMPRO constructs the corrected film as shown in Figure 5.

VII. Evaluation of the Error Correction Techniques

To facilitate sensitivity analysis, an option is provided ennabling the user to run three variants of TEMPRO with the same terrain history and the same noisy film. First, TEMPRO can be run with the standard conflict resolution and compatibility checks as described above (STANDARD). Second, TEMPRO can be run with the standard compatibility checks but with conflict resolution done by inversely ordering each list of possible configurations (REVERSED-PROBABILITIES). Third, TEMPRO can be run with standard conflict resolution but with no compatibility checks (NO-COMPATIBILITY-CHECKS).

Under a variety of conditions, the grades and performance indices achieved by STANDARD have been compared with those for the other two variants of TEMPRO, yielding some insight into the value of using temporal reasoning and conditional probabilities in conflict resolution. In eleven sample runs, STANDARD achieved a grade of .73, whereas REVERSED-PROBABILITIES (NO-COMPATIBILITY-CHECKS) had a grade of .58 (.68, respectively). (A grade of .5 might be expected by chance.) On the average, STANDARD chronologically backtracked one time and rejected 58 configurations, while REVERSED-PROBABILITIES backtracked 10 times as much and rejected 5 times as many configurations. STANDARD's grade advantage over NO-COMPATIBILITY-CHECKS (REVERSED-PROBABILITIES) is more (less) pronounced when the types are, roughly, equally likely.

The average grade, number of temporal backtracks, and number of rejected configurations in 93 runs of (STANDARD) TEMPRO (without also running (REVERSED-PROBABILITIES) and (NO-COMPATIBILITY-CHECKS)) were .81, .62, and 56. Analysis of these and other runs revealed that:

1. Compatibility check (1) ("Permanent objects never move"), as expected, caught more errors than the other two compatibility checks.
2. Compatibility checks 2 and 3 were more effective when there was a sparse distribution of unknown objects (<.2 expected per area).
3. Conditions (in combination) that overwhelm the compatibility checks (resulting in poor grades and performance indices) are large error rates (>.8 per area), dense unknown object distributions (>2 per area), and a large number of time periods (>10). For example, in one run with an error rate of .9 and a average density of 2 objects per area, TEMPRO received a grade of .76 (respectable) and a performance index of (16,4096) (poor). Further analysis of TEMPRO's performance (and details of the Franz LISP implementation) can be found in [Gumb, 1986].

One of the most promising system enhancements involves making the resolution of the inspector's sensor variable, so that the inspector's field of view could be carved more finely into as many as, say, 25 small areas instead of the current 9 large areas. The second and third compatibility checks (suitably modified) should become much more effective, and, with the resolution fine enough, the expected number of unknown objects per area might be plausibly restricted to a maximum of one.

The algorithm could be made more efficient by projecting into the future the number of permanent objects in each previously observed areas. Regarding extensions of TEMPRO (incorporating, for example, more types and more sophisticated compatibility checks), substantial changes in the underlying algorithm are required to guarantee that, in the general case, TEMPRO will produce *the most probable* corrected film.

Acknowledgements

Thanks go to Sarah Bottomley and Alex Trujillo for their work on implementing TEMPRO in Franz LISP, Arun Arya and Alex and Sarah for their assistance in preparing [Gumb, 1986], Ric Davis and Rick Craft for their advise on administrative as well as technical matters, Pierre Bieber for his suggestions on the present paper, and Gary Davidson, Christos Katsaounis, and Peter F. Patel-Schneider for lending a hand in the preparation of this paper.

References

[Durfee and Lesser, 1986] E. H. Durfee and V. R. Lesser. Incremental planning to control a black-board based problem solver. In *Proceedings of the Fifth National Conference on Artificial Intelligence*, pages 58–64, 1986.

[Durfee and Lesser, 1987] E. H. Durfee and V. R. Lesser. *Using Partial Global Plans to Coordinate Distributed Problem Solvers.* Technical Report 87-06, Computer and Information Science Department, University of Massachusetts, Amherst, MA, January 1987.

[Ferrante, 1985] R. D. Ferrante. The characteristic error approach to conflict resolution. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 331–334, 1985.

[Gumb, 1978] R. D. Gumb. Summary of research on computational aspects of evolving theories. *ACM SIGART Newsletter*, (67):13, 1978.

[Gumb, 1986] R. D. Gumb. *Filming a Terrain Under Uncertainty Using Temporal and Probabilistic Reasoning.* Technical Report 172, Computer Science Department, NMIMT, Socorro, NM, August 1986.

[Leblanc, 1981] H. Leblanc. Alternatives to standard first-order semantics. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic*, pages 189–274, Reidel, Dordrecht, 1981.

[Nilsson, 1986] N. J. Nilsson. Probabilistic logic. *Artificial Intelligence*, 28:71–88, 1986.