Chronicle Recognition in the WITAS UAV Project
A Preliminary Report

Fredrik Heintz
Dept. of Computer and Information Science
Linköping university, Sweden

E-mail: `frehe@ida.liu.se`

Abstract

This paper describes the chronicle recognition problem and reports its status in the WITAS UAV project. We describe how we use the IxTeT chronicle recognition system to define chronicles (scenarios or situations), like a vehicle passing another vehicle, and how it is incorporated in the WITAS architecture. We also discuss known problems with the current system and possible directions of future research.

1. Introduction

As the operator told the unmanned helicopter to watch the red Ford driving at high speed on the highway the helicopter did a sharp turn and increased its velocity to catch up with the speeding car, containing an escaped prisoner and his accomplices. As the distance decreased the operator got continuous updates of the actions of the fleeing vehicle: “reckless takeover with meeting traffic”, “turned left of the highway onto Arlington road”, “reckless takeover on the right side of the overtaken car”. While the helicopter watched the car and tried to anticipate the escape route the operator guided the police to set up a road block where the criminals could be caught and arrested.

This is only one of the possible scenarios that the WITAS[1] unmanned aerial vehicle (UAV) is supposed to handle. To manage this a number of technologies ranging over several research areas and disciplines are needed, for example autonomous navigation, planning for mission goals such as locating, identifying, tracking and monitoring different vehicle types, and construct internal representations of the observed world. Additionally, it should be able to identify complex patterns of behavior such as vehicle overtaking and traversing of intersections. The purpose of this long term basic research project, funded by the Wallenberg Foundation, is to research and develop technologies for the successful deployment of a fully autonomous UAV operating over road and traffic networks [5].

This paper describes and discusses the chronicle recognition subsystem which is responsible for recognizing complex scenarios such as overtakes, reckless driving and parking lot activities. The paper starts with an introduction to the chronicle recognition problem and situates it in previous research in Section 2. In Section 3 we describe the current status of the implementation of the chronicle recognition subsystem. In Section 4 some of the known problems with the current system are discussed and we outline how we intend to attack them. We also discuss other directions that the research could take. We end the paper with a summary in Section 5.

2. The Chronicle Recognition Problem

The problem we want to solve is to recognize complex scenarios, called chronicles, which is a set of temporally constrained events. This problem can be divided into two subproblems. The first problem is to find a representation to model the chronicles consisting of temporally related events that can be situated in a context. The second problem is to develop an inference mechanism that takes a stream of timestamped events and recognize which chronicle (scenario) instance was completed when.

We are currently using the IxTeT chronicle recognition system developed at LAAS in Toulouse, France, by Ghallab et al. [8] to do the modeling and recognition. IxTeT is based on temporal constraint networks and represent time as a linearly ordered set of discrete events. It is capable of representing constraints as before, equal, after and their disjunctions, and numerical constraints between time points in the form of intervals $[I^-, I^+]$ corresponding to the lower and upper bounds on the temporal distance between the time points, but not their disjunctions since then the problem be-

[1] Wallenberg Laboratory for Information Technology and Autonomous Systems

come NP-complete [4]. When a chronicle is recognized it can generate events that are fed back into the system to allow chains of events to be triggered. IxTeT, as any chronicle recognition system, can only recognize scenarios that have already happened, but it also keeps track of all possible developments (Ghallab calls this prediction).

One of the reasons we want to study chronicle recognition is to guide the planning and the execution of plans for the WITAS UAV. The purpose of this guidance is to improve the quality of the plans and the efficiency of the plan execution. Another reason is to reduce the amount of data sent to an operator of the UAV, instead of sending an endless stream of data like "car A at position $\langle x, y, z\rangle$ moving in the direction d" the operator can tell the UAV to monitor overtakes, reckless driving, vehicles passing a certain location or building or all activities made by a certain vehicle.

Other applications of chronicle recognition is in the surveillance of dynamic systems, it has for example been used with success to monitor gas turbines [1] and telecommunication networks [2].

Other similar approaches to IxTeT have been proposed. One example is Fontaine and Ramaux [7] who tries to match temporal constraint networks, instead of satisfying them, by constructing one graph for the chronicle being recognized and one for the actually observed events, called the session, and trying to match them. An approach close to text pattern recognition techniques using finite-state automaton whose transitions correspond to the observed events is presented by Lévy [10]. His approach is very efficient in recognizing sequential chronicles while other structures impair the performance.

A related research area is plan recognition where you infer what plan an agent is executing by observing the agents' interactions with the world and by maintaining a model of the mental states of the agents, which is a much more general concept than a chronicle or scenario [9]. According to Pynadath and Wellman [12] the difference between general pattern recognition (to which you can count chronicle recognition) and plan recognition is that plans are made by a rational agent with a mental state which could be used to improve the matching. In chronicle recognition you only take the actual events into account not why the agent causing the events did them. Another difference is that plan recognition systems usually do not deal explicitly with time, instead they focus on the plans and what actions or sub plans they are made of.

3. Current Status

In order for the chronicle recognition system to detect a situation, like a car passing another car, there are three tasks that needs to be done: a set of primitive events have to be defined and implemented (i.e. detected), a detailed definition of the situation have to be written and finally the definition has to be encoded in the language of the chronicle recognition system. The primitive events can be (and usually are) the same for all situations within a certain domain, but each domain and implementation of it has its own set of primitives. To find and define them is a hard knowledge engineering task. In our domain we started with what we knew we will get from the sensors, mainly the vision system, and then saw what events we could generate from the data. In our case the sensor data only provide us with information like "the red blob is at position $\langle x, y, z\rangle$", "the red blob might be a Ford", "the direction of the red blob is d" and "the speed of the red blob is s km/h". From this basic information together with a geographical information system (GIS) we can generate events such as car A is on lane L on the road segment R, and car A is driving the wrong direction of the road. We also calculate qualitative information like car A is beside car B and car A is close to car B.

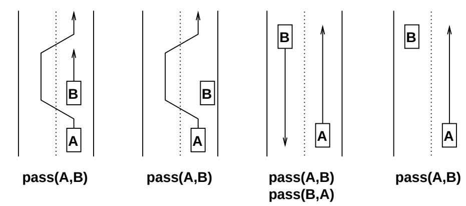

Figure 1. Typical pass situations.

One of the implemented chronicles is the pass chronicle. Our definition of a pass is: "A vehicle A passes another vehicle B if(f) A is moving from a position where B is in front of A to a position where A is in front of B and vehicle A at some point in time is beside vehicle B, and both A and B are on the same road (or in the same intersection)." A sketch of four typical situations belonging to this definition is shown in Figure 1. The implementation of the pass definition is shown below.

```
-- Recognizes car ?c1 passes car ?c2
chronicle pass[?c1, ?c2]
{
  -- move[?c1] should be true during the
  -- whole chronicle i.e. it should be true
  -- at time point before and not change
  -- before time point after
  event(move[?c1]: (?, true), before)
  noevent(move[?c1]: (true, false),
          (before+1, after))

  -- the constraints on the relative
  -- position of the cars
  event(in_front_of[?c2, ?c1]: (?, true), t0)
  event(beside[?c1, ?c2]: (?, true), t1)
  event(in_front_of[?c1, ?c2]: (?, true),
```

```
        after)

  -- The requirement that the vehicles
  -- be on the same road the whole time
  event(on_road[?c1, ?r]: (?, true), before)
  noevent(on_road[?c1, ?r]: (true, false),
          (before+1, after))
  event(on_road[?c2, ?r]: (?, true), before)
  noevent(on_road[?c2, ?r]: (true, false),
          (before+1, after))

  -- The temporal constraints encode that
  -- this is serial chronicle
  before <= t0
  t0 < t1
  t1 < after
  t1 - t0 in [1, 20]
  after - t1 in [1, 30]
}
```

3.1. The Chronicle Recognition Architecture

The chronicle recognition system needs two types of inputs to do its job. First of all, it needs to know what to recognize, i.e. which chronicles should it match against when the primitive events arrive. Second, it needs time-stamped observations, or primitive events, to do the actual recognition. Figure 2 shows the setup we use in the current WITAS architecture. The subsystems shown in the picture are:

- CONTAP: Which is the reactive procedural system responsible for controlling the UAV and also the active vision system. The CONTAP is also responsible for turning the chronicles being recognized on and off.

- The vision system which is responsible for taking input from the cameras and extract primitive information such as what cars are seen and their properties such as position, direction and velocity. This information is not very accurate and there is also the problem of anchoring the image data received from the vision system to actual vehicle objects in the knowledge base.

- DOR: the Dynamic Object Repository which is responsible for storing the dynamic data used by UAV. It contains the information extracted by the vision system and information about the UAV itself, like its position, velocity, heading and so on.

- GIS: the Geographical Information System which is responsible for keeping track of the static geographic data needed by the UAV and for making geographic calculations like calculating line of sight and possible paths in the road network.

- Front End: which is responsible for gathering the currently interesting pieces of data from the DOR and the GIS and combine them into the primitive events that the chronicle recognition system uses. Examples of primitive events are "car A is on road R" `on_road[A, R]`, "car A is moving" `move[A]`, and "car A is beside car B" `beside[A, B]`.

- Chronicle Recognition Engine: which is responsible for recognizing the chronicles which the CONTAP tells it to recognize. The primitive events generated by the front end and already recognized chronicles are used as inputs. The recognized chronicles are reported to the CONTAP.

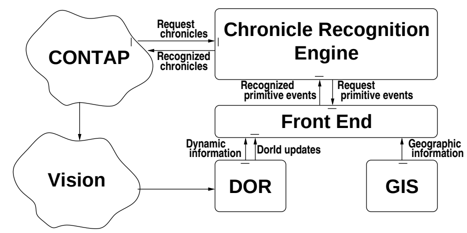

Figure 2. The data flow in the WITAS chronicle recognition system.

A prototype of the chronicle recognition engine subsystem is implemented and is working. It is not based on the original IxTeT implementation but rather a new implementation made by Christophe Dousson for France Telecom called CRS [3]. The main reason being that it is designed to be embedded in other applications, which IxTeT was not. We are currently working on integrating it with the rest of the WITAS architecture as depicted in Figure 2.

4. Further Work

When the chronicle recognition system is fully integrated in the WITAS architecture we intend to make extensive tests to see what problems exists and what needs to be done in order for the system to be more useful than it already is. We will also be able to refine the chronicles as we see what situations they can and can not recognize.

One of the known problems with the current system is that it has no explicit support for vagueness in the event descriptions, cannot reason explicitly about the certainty of future events, and depend on the completeness of the event detections (i.e. all relevant changes in events have to be detected) for the completeness of the chronicle recognition. Some of these problems could be solved by writing specialized chronicles that explicitly takes missing observations into account but it is very easy to forget some special cases and the real-time properties of the chronicle recognition system is dependent on the number of chronicles being recognized which reduce the usefulness of the work-around.

Therefore the top priority is to extend the modeling and reasoning capabilities to take uncertainty into account. There are several different types of uncertainty that needs to be accounted for, for example false, noisy or missing observations, and noisy or incomplete domain models.

In the longer perspective some of the possible directions of future work are:

- Extend the temporal expressivety based on the work done on temporal constraint networks [4, 11, 13].
- Use learning to construct chronicles based on statistical data about the domain being modeled [6].
- Extend the expressivety of the representation towards more general plan recognition. This would at least include some constructs for handling the agents being observed and their mental state. This will probably be needed if we want to model scenarios like "car A is heading for X", "car A is trying to escape", or other scenarios that involves the will or intention of the drivers of the vehicles.

5. Summary

In this paper we have discussed the chronicle recognition problem, which is a way of modeling and recognizing situations or scenarios, called chronicles, based on temporally related events. We have described how a chronicle recognition system called IxTeT, and an implementation of it called CRS, is being used in the WITAS UAV project to detect chronicles like vehicles overtaking and passing other vehicles. We have also discussed some of the known problems of the current system and possible directions for future research. The experience with the current prototype is positive and more results are expected as the system is further developed and tested. The aim of the project is to be able to handle uncertainty and incomplete observations while retaining the efficiency of the current system.

Acknowledgments

This work is funded in part by the WITAS project grant from the Wallenberg Foundation, Sweden, and part by the ECSEL/ENSYM graduate studies program.

References

[1] J. Aguilar, K. Bousson, C. Dousson, M. Ghallab, A. Guasch, R. Milne, C. Nicol, J. Quevedo, and L. Travé-Massuyès. TIGER: real-time situation assessment of dynamic systems. *Intelligent Systems Engineering*, pages 103–124, 1994.

[2] S. Bibas, M.-O. Cordier, P. Dague, C. Dousson, F. Lévy, and L. Rozé. Alarm driven supervision for telecommunication networks: I - off-line scenario generation *and* II - on-line chronicle recognition. *Annals of Telecommuncations*, pages 493–508, 1996.

[3] The Cronichle Recognition System C.R.S. web-page. http://crs.elibel.tm.fr/en/index.html. verified January 11th 2001.

[4] R. Dechter, I. Meiri, and J. Pearl. Temporal constraint networks. *Artificial Intelligence*, 49:61–95, 1991.

[5] P. Doherty, G. Granlund, K. Kuchinski, E. Sandewall, K. Nordberg, E. Skarman, and J. Wiklund. The witas unmanned aerial vehicle project. In W. Horn, editor, *ECAI 2000 Proceedings of the 14th European Conference on Artificial Intelligence*. IOS Press, 2000. to appear.

[6] C. Dousson and T. Vu Duong. Discovering chronicles with numerical time constraints from alarm logs for monitoring dynamic systems. In D. Thomas, editor, *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI-99-Vol1)*, pages 620–626, S.F., July 31–Aug. 6 1999. Morgan Kaufmann Publishers.

[7] D. Fontaine and N. Ramaux. An approach by graph for the recognition of temporal scenarios. *IEEE transactions on System, Man and Cybernetics*, 1997.

[8] M. Ghallab. On chronicles: Representation, on-line recognition and learning. In L. C. Aiello, J. Doyle, and S. Shapiro, editors, *Proceedings of the Fifth International Conference on Principles of Knowledge Representation and Reasoning*, pages 597–607, San Francisco, Nov. 5–8 1996. Morgan Kaufmann.

[9] H. A. Kautz and J. F. Allen. Generalized plan recognition. In T. Kehler and S. Rosenschein, editors, *Proceedings of the Fifth National Conference on Artificial Intelligence*, Los Altos, California, 1986. American Association for Artificial Intelligence, Morgan Kaufmann.

[10] F. Lévy. Recognising scenarios: a study. In *Proceedings of the Fifth International Workshop of Diagnosis*, pages 174–178, 1994.

[11] P. H. Morris and N. Muscettola. Managing temporal uncertainty through waypoint controllability. In D. Thomas, editor, *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI-99-Vol2)*, pages 1253–1258, S.F., July 31–Aug. 6 1999. Morgan Kaufmann Publishers.

[12] D. V. Pynadath and M. P. Wellman. Accounting for context in plan recognition, with application to traffic monitoring. In Besnard, Philippe and S. Hanks, editors, *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence (UAI'95)*, pages 472–481, San Francisco, CA, USA, Aug. 1995. Morgan Kaufmann Publishers.

[13] T. Vidal. Controllability characterization and checking in contingent temporal constraint networks. In A. G. Cohn, F. Giunchiglia, and B. Selman, editors, *Proceedings of the Conference on Principiles of Knowledge Representation and Reasoning (KR-00)*, pages 559–570, S.F., Apr. 11–15 2000. Morgan Kaufman Publishers.