Enriching a Semantic Network Language by Integrating Qualitative Reasoning Techniques

T. Fuhr, F. Kummert, G. Sagerer*

Technische Fakultät, Angewandte Informatik
Universität Bielefeld, Postfach 100131, 33501 Bielefeld, Germany
e-mail: fuhr@techfak.uni-bielefeld.de

Abstract. To interpret sensor signals like images, image sequences, or continuous speech the representation and use of task-specific knowledge is necessary. The paper scetches a framework for the representation and utilization of declarative and procedural knowledge using a suitable definition of a semantic network. To meet the needs of machine-human interaction we extend this framework in two ways. A temporal model similar to Bruce is incorporated and representational structures are integrated to formulate qualitative (relational) knowledge. The problem-independent inference rules are extended to allow for the temporal prediction and the dynamic refinement of this knowledge. Our integration of relational knowledge exemplarily shows how semantic network representations can benefit from developments in qualitative reasoning research.

1 Introduction

The KL-ONE-like semantic network language ERNEST [14, 18] has been especially designed for the purpose of knowledge–based signal understanding and has proven successfully in different larger scale applications in the area of image and speech understanding. Because of this quality, this semantic network language shall now also be used to model the domain–specific knowledge needed by a knowledge–based system that communicates with a human user via continuous speech and disposes of a camera input. However, this extended task domain introduces a new issue: the communication with a human user makes it essential to allow for the use of qualitative terms in this communication.

A lot of expertise on the modeling and utilization of qualitative knowledge has been evolved from qualitative reasoning (QR) research. Beside the research in qualitative temporal [3, 1, 8] and spatial reasoning [10, 11, 4] a lot of progress has been achieved in the field of qualitative physics and qualitative simulation [19, 4]. Also, QR research addresed issues regarding the interaction of numerical data and qualitative models [12, 7, 5].

In this paper, we present an extension of the semantic network language ERNEST that fulfills the above mentioned needs and is designed in awareness of the approaches developed in QR research. Our extension allows the formulation of qualitative knowledge in the semantic network model in a way, that firstly, supports the construction of qualitative interpretations from numerical input sensor

* This work has been supported by the German Research Foundation (DFG) in the project SFB 360.

data on the basis of this model. Secondly, it allows to use qualitative expectations (e.g. some spatial position of an object qualitatively described in a sentence uttered by the human user) to infer focused expectations on the numerical input (e.g. excluding parts of an image frame, where this object need not be searched).

We furthermore outline how a notion of time can be settled into the formalism, that accounts for both, the sampledness of the analyzed data and the necessity to formulate complex temporal relations between concepts. Temporal prediction is introduced as a new inference rule.

In the next section, the paper describes a subset of the ERNEST language. In section 3 we present the new representational structures to represent qualitative knowledge together with the adapted inference rules. Section 4 outlines our integration of an explicit notion of time into the network. A small example is explained in section 5 that illuminates one aspect of our extension of ERNEST by especially focusing on the interaction of qualitative and numerical data. Finally, we give some conclusions and an outlook on further work.

2 The Semantic Network Language

ERNEST is a semantic network language that has been designed to meet the specific needs of pattern interpretation and understanding tasks. In this section we describe only that subset of its representational vocabulary that is essential to understand the rest of this paper. Beside other details, we particularly omit how specialization hierarchies can be formulated, in which way inheritance is bound to this hierarchy and how knowledge can be organized in different levels of abstractions w.r.t. the signal level. The reader will find descriptions of the complete ERNEST language in [16, 18, 14].

As usual in semantic networks, **concept** nodes are the central representational entities to model notions of the task domain. In ERNEST a **concept** is an intensional description of some notion and can be specified by establishing **links** to other concepts as well as by annotating it with **attributes** and **relations**. Finally, the definition of a concept must be completed by defining a **judgment function** that allows to estimate the correspondence of some area of the sensor signal to the notion represented by the concept. This correspondence cannot simply be characterized as being true or false due to the different certainty, quality, and reliability of the input sensor data. The judgment calculus underlying the judgment functions can be determined by the modeler. A concept can be specified as the composition of other concepts by establishing **part** links to other concepts (e.g. Car $\xrightarrow{part}$ Wheel). **Attributes** (e.g. color, size) can be used to represent features characterizing the notion modeled by the concept. For every attribute A of a concept C a function f for the **computation of values** (of A) must be defined. Attributes of C and attributes of its parts may be used as parameters of f. Also a function for the **inverse computation of values** (w.r.t. f) can be bound to A. This allows to express in which way attributes of a concept can restrict the attributes of its parts. The possibility to bind procedural knowledge to attributes allows the modeler to integrate domain-specific algorithms (e.g. for signal filtering, segmentation) appropriate for handling the kind of the domain-specific

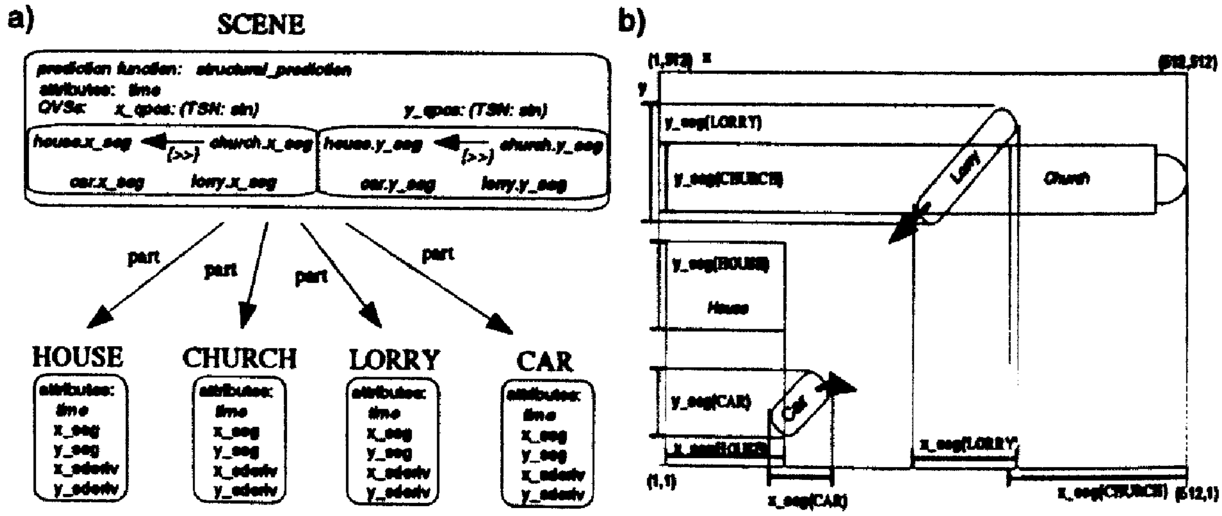

Fig. 1 Simple semantic network model of some image scene

sensor data. Finally, a **judgment function** for an attribute may be specified. However, a concept is usually not definable as simply *any* combination of its parts and also not *any* combination of values of the concept's attributes will correctly characterize it. Hence, a concept may be annotated by **relations** (e.g. "height < length") to constrain such combinations. A relation is defined by a **judgment function** that can take attributes of the concept and its parts as parameters. That relations are being *judged* rather than being *tested* on being true or false is a concequence of the fact that attributes and parts are judged. Again, an inverse function (**inverse judgment of relation**) can also be supplied to model the restrictions imposed on attributes of parts due to the semantics of the relation.

Fig. 1-a) shows a very small ERNEST network to model simple scenes that contain a church, a house and the two moving objects car and lorry (Fig. 1-b)). All objects are attributed by bounding rectangles, given as the two projection segments w.r.t. to the x and y axis (x_seg, y_seg). Furthermore, their velocity is qualitatively described through the signs of the derivatives of the segments' bounds (x_sderiv, y_sderiv). All network entries printed in italics, belong to the extended network language and will be explained later.

To allow the utilization of the modeled knowledge the ERNEST system embodies a set of problem–independent inference rules. The basic idea behind these rules is, that the interpretation of some signal area is established by connecting it to some notion (i.e. concept) in the knowledge base. Such connections are represented by **instance** nodes. An instance describes an extension of the corresponding concept. The concept's parts are replaced by instances of parts, concrete values are given for all its attributes, and also judgments for attributes, relations and the instance itself. The latter judgment expresses the certainty (and/or validity, quality etc.) of the assumption that the corresponding signal area is correctly interpreted by the related concept. In an intermediate state of the analysis it may occur that for some concepts instances cannot be computed due to missing parts. Nevertheless, the available partial information can be used to constrain an uninstantiated concept, i.e. derive restrictions for its attributes. The third node type **modified concept** allows to represent constrained but uninstantiated con-

IF	**for a concept A or a modified concept $M_j(A)$ instances exist for all parts**
THEN	**create instances $I_k(A)$ as follows:** **• create for $I_k(A)$ an empty instance,** **• connect $I_k(A)$ with those instances referred to by the premise,** **• activate the attached functions for $I_k(A)$ in the sequence: judgment of links, computation of attributes, *computation of QVSs, extraction of QVSs from members, qualitative propagation on QVSs, member-centered propagation for each attribute of A which is a member of a QVS of A,* judgment of attributes, judgment of relations, judgment of the concept A**

Fig. 2 Rule for the **creation of instances**

cepts. In modified concepts only **restrictions** (i.e. sets of values that are still admissible) are given for attributes, and only some parts (given as instances or modified concepts) may be bound.

The rule for the creation of instances (see Fig. 2) shows how instances are constructed by the ERNEST system. It reflects the idea that the recognition of some complex object in the data needs the detection of all its parts as a prerequisite. It can also be seen that due to this rule concepts with no parts can be directly instantiated on the basis of the sensor data, and that increasingly complex concepts can be constructed in a data-driven fashion. Furthermore, a rule for the **data-driven modification of concepts** is defined that looks very similar to the instantiation rule. The important difference is that not all parts must be bound to a given concept (or modified concept) and only restrictions may be given for attributes. So, this rule formulates a data-driven propagation based on *partial information*. Contrarily to instances, a **model-driven modification of concepts** is also possible (see Fig. 3). Here the inverse functions come into play. The ERNEST system includes a problem-independent algorithm that controls the activation of the inference rules to construct interpretations for the input data. Due to the noise inherent in sensor data several competing interpretations may be inferred for identical signal areas. Hence, the most adequate interpretation must be searched for. The control algorithm manages this in an A*-based fashion. For details see [16, 14].

The successful application in different task-domains indicates the quality of the ERNEST language. The applications cover the detection of a roboter hand in complex scenes [13], the diagnostic interpretation of image sequences of the heart [17], and the understanding of spoken language in a speaker-independent dialog system [18]. The obtained results show that the network language and the problem-independent control algorithm are able to handle totally different applications in an efficient manner.

3 Incorporating Qualitative Knowledge

The incorporation of qualitative knowledge must account for its *dynamic* nature. Spatial relations between objects in a frame t are usually not (totally) known in advance and hence, cannot be modeled by static relations. The spatial structure

IF	for a concept A or a modified concept $M_j(A)$ a new instance I(B) or a new modified concept M(B) exists and there is a link $B \xrightarrow{part} A$
THEN	create new modified concepts $M_k(A)$ as follows: • create for $M_k(A)$ a new empty modified concept, • connect $M_k(A)$ to all instances and modified concepts referred to by $M_j(A)$, • activate for $M_k(A)$ the attached functions of A and B in the following sequence: ∘ inverse computation of attributes of B, which have an attribute of A as an argument, ∘ *inverse computation of QVSs of B, which have an QVS of A as an argument,* ∘ *member–centered propagation for each attribute of A which is a member of a QVS of B* ∘ inverse judgment of relations of B, which have an attribute of A as an argument, ∘ inverse judgment of links of B, which have A as the goal node, ∘ functions of A like for the creation of instances

Fig. 3 Rule for the **model–driven modification of concepts**

of the objects in frame t may be different from their relations in frame $t+1$ and might even constrain the latter. Furthermore, some spatial description extracted from an utterance could constrain the image positions (pixel areas) for the mentioned objects. Consequently, relational knowledge should be extractable from numerical or other lower level data (data-driven usage). Additionally, it should be able to constrain lower-level numerical and qualitative data (model-driven usage), and also should be usable to derive temporal expectations (predictive usage). However, in the ERNEST system only *static* expectations on attributes can be modeled via the **relation** entry in concepts. Those relations may constrain attribute values (model-driven usage) but their fulfillment can only be judged (data-driven usage). Hence, an extension of the ERNEST system is necessary.

3.1 New Representational Structures

To formulate dynamic qualitative knowledge we introduce **base relation sets** (BRSs) and **temporally structured neighborhoods** (TSNs) as representational entities that have network-wide validity. Their content can be used to specify the **qualitative value spaces** (QVSs) that can additionally be formulated in each concept.

Base relation sets. The user can define sets of binary base relations for any type of attributes he uses in the semantic network. So e.g. point relations may be introduced, Cohn's 2D relations ([4]), etc. A **base relation** r is described by two procedures. The first procedure tests for the fulfillment of the relation using the restrictions/values of two attributes. In this way the relations can be numerically defined. The second procedure propagates the restriction/value of one attribute to another according to r. Finally, for each set of base relations a procedure implementing its composition table must be given. Fig. 4-a shows the base relation set segmentrels. The relations are adaptations of Allen's relations to a)

I --r--> J		definitions		J --r--> I
lower-disjoint	<<	$g(I) < s(J)-1$	>>	upper-disjoint
meets	m	$g(I) = s(J)-1$	mi	met-by
overlaps	o	$s(I) < s(J) \le g(I) < g(J)$	oi	overlaped-by
starts	s	$s(J) = s(I) \le g(I) < g(J)$	si	started-by
included	i	$s(J) < s(I) \le g(I) < g(J)$	c	contains
finishes	f	$s(J) < s(I) \le g(I) = g(J)$	fi	finished-by
equals	e	$s(J) = s(I) \le g(I) = g(J)$	e	equals

b)

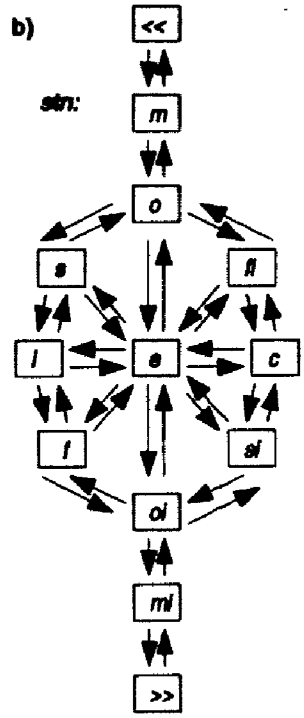

Fig. 4 Binary relations on 1D spatial segments and their TSN

one-dimensional (spatial) segments. A *segment* I is a chain of consecutive discrete points with smallest point s(I) and greatest point g(I). In our case, the points correspond to pixel positions w.r.t. a single image axis, and consecutive points have difference 1. The corresponding composition function implements Allen's transitivity table.

Temporally structured neighborhoods. On sets of relations often continuity structures, called *neighborhoods* (Freksa) can be imposed ([8, 11, 4]), that reflect and depend on the possible valid transformations on the related attributes. Furthermore, neighborhoods support the qualitative prediction of relations ([4, 9]). We incorporated a special version of neighborhoods into the semantic network, called **temporally structured neighborhood**. They enable a qualitative temporal characterization of changes analogous to Forbus' *equality change law*. Consequently, a temporally more precise qualitative prediction is achievable when simultaneous changes may happen (see [9]). Fig. 4-b shows the TSN stn we imposed on segmentrels. Its structure reflects that projections of image regions scale *and* translate at simultaneously. Furthermore, due to the discreteness of segments all transitions between relations must be regarded as being of non-zero duration (solid arcs).

Qualitative value spaces. In each concept an arbitrary number of relational networks, called **qualitative value spaces**, can be formulated. A QVS q of a concept C has an entry referencing a single TSN. QVS nodes, called *members*, are identified with attributes of C or of its parts. Arcs may be labeled by disjunctions of base relations of the referenced TSN. If any relation of the base relation set may hold between two members, no arc is established. A QVSs **restriction** or **value** is given by some refined version of its relational network. Finally, two user procedures may be associated with a QVS q. The **QVS computation function** can take QVSs of C or QVSs of C's parts to compute a refined restriction or a value of q. The **inverse computation function of a QVS** supports the refinement of QVSs of C's parts. Both functions directly counterpart the functions bound to attributes and allow e.g. the incorporation of transformations like Metric–to–Allen and Allen–to–Metric [12].

Fig. 6-a shows the QVSs represented in the concept SCENE. In the extended ERNEST version. They express the a priori known spatial relation between the stable landmark objects Church and House in a Guesgen-like fashion [10]. They also express that a Car and a Lorry with unknown qualitative position are expected in the scene. Restrictions and values of these QVSs that were computed during an example analysis process can be seen in Fig. 6-b,c,d,e (see section 5).

3.2 Adapted Inference Rules

Three general system procedures are implemented and incorporated into the semantic networks' inference rules (see Figs. 2, 3). Our extensions of the rules are printed in italics: **computation and inverse computation of QVSs** (see section 3.1), **extraction of a QVS** from its members, **qualitative propagation** on QVSs, and **member-centered propagation.** In this way, the utilization of the dynamic qualitative knowledge within QVSs is ensured. This is also demonstrated in our small example in section 5.

Extraction of a QVS. This routine applies the test routines bound to base relations in the following way to refine QVSs. For each pair of members a and b of some QVS q determine the base relations that may hold between them by testing for all base relations of the corresponding BRS. Intersect this set with the current label between a and b to yield the new label. In this way QVSs can be computed bottom-up from (numerical) attribute data and numerical data can be checked for accordance with QVSs.

Qualitative propagation. This is a slightly modified Allen-like propagation ([1]) algorithm that works on QVSs: the initial queue contains *all* QVS arcs that were changed during the last extraction process. So, the extraction phase is clearly seperated from the propagation. This system routine uses the composition functions bound to base relation sets. The qualitative propagation is always activated directly after the extraction phase (see Fig. 2). So data-driven extracted QVSs can be refined, due to the semantics of the underlying BRS.

Member-centered propagation. The refined qualitative constraints in a QVS can be used to further restrict attribute restrictions of the modified concept or instance or to propagate numerical restrictions top-down to parts. The new restriction for a member attribute A is calculated simply, by intersecting all numerical restrictions propagated along each arc leading to A with the actual restriction for A. The restriction obtained along an arc is given by the union of the restrictions returned by the propagation procedures bound to the base relations labeling this arc (see description of BRSs). This kind of propagation is a simplified numerical propagation [15] in the sense that information is transfered to a member from its direct QVS neighbors only and resulting changes are not further propagated to other members. Although being computationally less expensive, this propagation doesn't yield the most restrictive result for the regarded member. However, most often the less refined attribute restrictions already allow to sufficiently focus expectations on lower level concepts.

4 Introducing Time into the Semantic Network

We decided to introduce a model of time similar to Bruce [3] since in opposite to [1, 8] it accounts for both, the sampledness of the data and the need to formulate complex temporal relationships. Bruce regards time as a set of time-points. Gapless chains of linearly ordered time-points are called **time-segments**. Binary relations, similar to Allen [1], are defined on time-segments that allow the formulation of qualitative temporal knowledge. Consequently, we annotate each concept with an attribute time that can take a time-segment as value. As a standard semantics we incorporate computation and inverse computation functions for time that ensure, that each concept's time value expresses which subsequence of the signal data it interpretes. The attributes time can be treated by the modeler like any other attribute. So, qualitative relations between time-segments can be introduced and used like other qualitative knowledge according to section 3.

An adequate tool for the analysis of image/utterance sequences needs the capabality to express expectations on future frames due to the (intermediate) analysis results of the previous images/utterances. This expectations have a highly dynamic character and do not only affect attribute values, but might also affect the bindings between modified concepts that interpret subsequent data. For this purpose we introduce a new inference rule that allows for inferring expectations along the time axis: **rule for the creation of corresponding modified concepts** (see Fig. 5). This rule is not directly executed by the problem-independent control algorithm [14]. It rather must be activated by a user prediction function. These user functions may be bound to concepts via the newly introduced optional slot **prediction function**. In contrast to the attribute and QVS functions (see section 3), they can use information of the whole subgraph underneath the concept they belong to. This together with the optionality of the prediction function entry in concepts allows for the implementation of centralized temporal predictions: only one high-level concept (e.g. SCENE) embodies a prediction function, whenever it is instantiated for some time-segment $[t, t]$, this function uses the full scene description (i.e. including all instantiated subconcepts) to derive expectations for the scene in the next frame. On the other hand, there might just as well concept specific prediction functions be defined in each concept. In this way, a decentralized local temporal prediction can be formulated.

The modeler is offered a **qualitative prediction** routine as a tool to support him in realizing temporal prediction functions. Given a set of QVSs and qualitative derivative values for their members the algorithm determines new arc labels for each given QVS on the basis of their TSNs, assuming that all changes occur simultaneously. The new label sets express which relations are principally admissible between the connected members in any temporally directly subsequent qualitative state. If the *simplest action assumption* ([6]) can be applied to the signal data the new labels express the qualitative expectations on the data of the subsequent sample point (for further details see [9]). The prediction algorithm is

IF for a concept A an instance $I_j(A)$ or a modified concept $M_j(A)$ exists
THEN create new modified concepts $M_k(A)$ as follows:

- create for $M_k(A)$ a new empty modified concept,
- insert restrictions for all attributes and all QVSs due to the prediction function of A,
- connect $M_k(A)$ to instances and modified concepts selected by the prediction function,
- connect $M_k(A)$ to $I_j(A)$ or $M_j(A)$, respectively, via the correspondence link,
- activate the attached functions for $M_k(A)$ like in the rule for the creation of instances (see Fig. 2)

Fig. 5 Rule for **creation of corresponding modified concepts**

offered to the user as a tool he may use to implement user prediction functions (see previous section). We are well aware, that Forbus' assumption will often not be fulfilled by image sequence data. Furthermore, in contrast to [5] we are not able to compute a total envisionment of the observed system prior to the analysis to be able to fill observation gaps during the analysis. The possible behavior of the scene objects captured in images are usually to manyfold. Hence, what is needed is an adapted qualitative prediction algorithm that also predicts the relations of non-directly successive states that could match the next image frame. So far, we have no solution for this.

In connection with this new inference the new link type **correspondence** between modified concepts and/or instances is introduced. It expresses that the connected modifications or instances refer to the ontologically same object in the world, although they interpret it with respect to different time-segments.

5 A Small Example

The example analysis process we scetch in this section is based on the very simple semantic network model of some image scene shown in Fig. 1-a) (including the parts printed in italics) and no real image data was used. However, it is sufficient to describe, how the interaction between numerical attribute data and qualitative knowledge works due to our adapted inference rules.

We omit to regard any competing modified concepts or instances, hence we can ignore judgment computation. To visualize the effects of the inference rules we depict the bounding boxes that characterize the modified concepts and instances that are currently computed for the objects. Since the concept SCENE contains QVSs only but no attribute or relation entries, the application of the modification rules as well as the instantiation of the concept SCENE merely consist of the activation of the newly introduced functions (cf. section 3.2). So, their effect can purely be demonstrated. Our example analysis starts at a point where the content of the image frame (scetched in Fig. 1-b)) taken at time point t is fully interpreted by the construction of I_1(SCENE). The bounding boxes in Fig. 6-a) capture the objects' positions and the QVSs x_qpos and y_qpos exactly represent their spatial relations seperately for the x and y axis (cf. [10]). The little

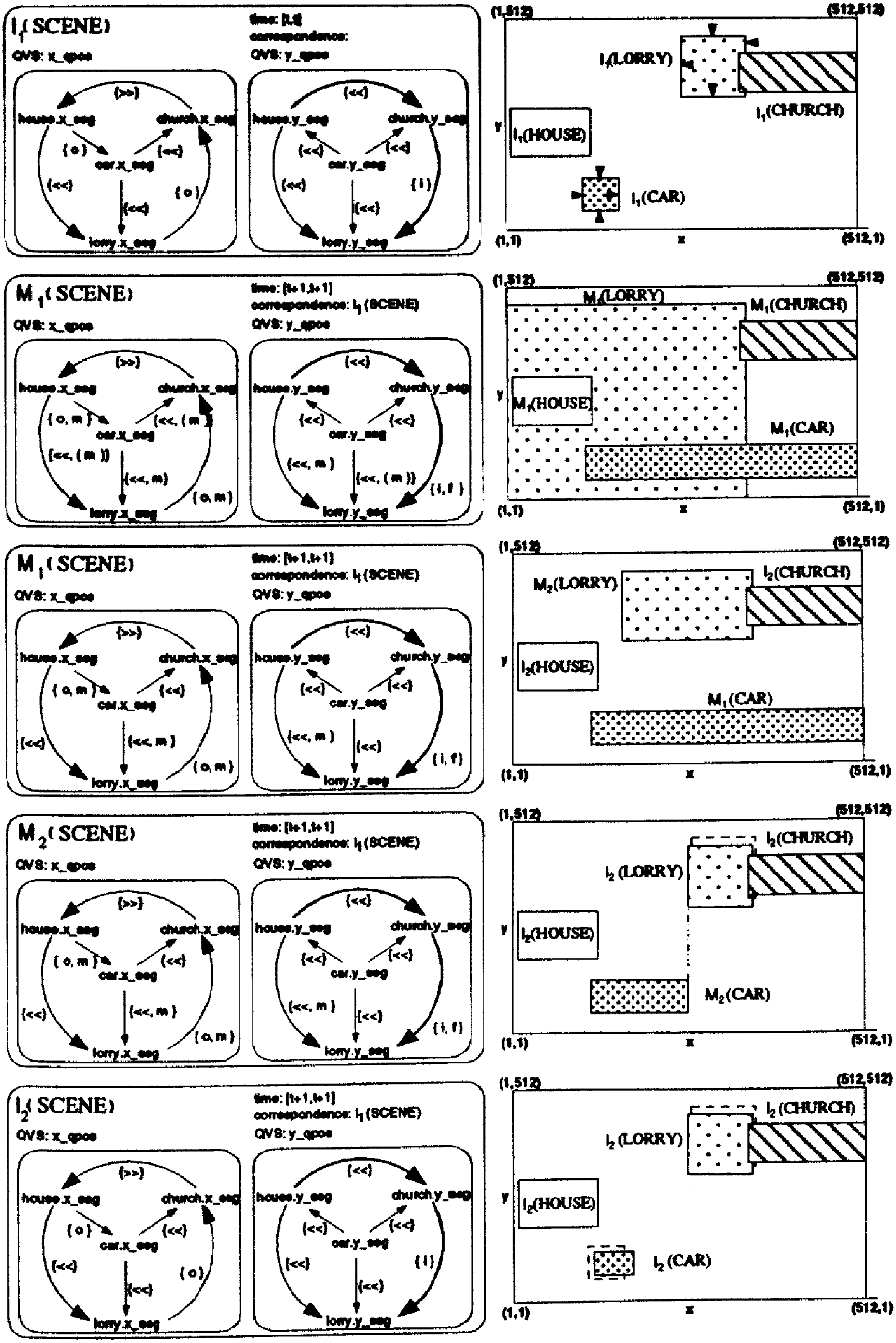

Fig. 6 The SCENE concept and some analysis states

arrows at the rectangle for CAR and LORRY represent their qualitative velocity. By activating the prediction function structural_prediction of SCENE concept-local predictions w.r.t. time point $t+1$ are computed for each instance of time segment$[t,t]$. The new QVSs in Fig. 6-b) including the **meets** relations in round brackets result from applying our qualitative prediction algorithm to the QVSs of I_1(SCENE) (Forbus' simplest action assumption is assumed to hold.). The subsequent qualitative propagation eliminates these **meets**. The restrictions for the objects at time-point $t+1$ are computed using the qualitative velocity information only. Church and house are regarded as immovable. It should also be noted that the prediction rule connects the modified concept M_1(SCENE) with I_1(SCENE). Next, a model-driven modification of LORRY takes place yielding the refined box of M_2(LORRY) in Fig. 6-c). This box is computed by the member–centered propagation function. Subsequently, LORRY is instantiated directly from the data (its old position is illustrated by the dashed box), SCENE, is modified data-driven (the QVSs don't change), and CAR is modified model-driven. The result of these three steps shows Fig. 6-d). CAR's bounding rectangle is substantially refined. After activating the instantiation of CAR all prerequisites are given to instantiate SCENE to I_2(SCENE). As part of the instantiation rule the QVS extraction function restricts all arc labels to a single relation. Fig. 6-e) shows the resulting interpretation for the frame at time segment $[t+1,t+1]$.

This example shows that our extension of the network formalism allows to

- focus the search space for instances by allowing to convert qualitative knowledge top-down to numerical expectations (this is an important aspect also, when a human user verbally describes the position of some object to be identified by the image analysis system),
- extract qualitive knowledge from the numerical sensor data, and
- derive qualitative expectations on temporally successive sensor data.

We further want to point out, that this example is not a claim for a Guesgen-like ([10]) representation for modeling spatial properties of complex scenes. This representation was only chosen in this paper, because it is easy to understand without much explanation.

6 Conclusion and Outlook

We scetched the semantic network language ERNEST that has already proven its quality for the knowledge–based understanding of sensor data, like images and speech. The incorporation of a notion of time is described. Furthermore, representational structures are introduced to model qualitative (relational) knowledge. The problem–independent inference rules of ERNEST are extended to allow for the dynamic refinement of this knowledge and its interaction with numerical data. We believe that our extension is a further example of a beneficial integration of qualitative reasoning into another reasoning framework, namely a semantic network language. With our extension arbitrary domain-specific relations between concepts can be modeled in a systematic way without violating the demands e.g. put forward in [2]: they have a well-defined domain-specific semantics and can be utilized by domain-independent algorithms.

In our future work we will focus on the integration of judgments for QVSs and their labeling relations. This is necessary since relations that have continuous domains in the world have to be modeled on the basis of sampled input data. Also an adaptation of our prediction algorithm is of great interest, that is based on restrictions less strong than Forbus' *simplest action assumption* .

Literatur

1. J. F. Allen. Maintaining knowledge about temporal intervals. *CACM*, 26(11):832–843, November 1983.
2. R. J. Brachman. On the epistemological status of semantic networks. In N. V. Findler, editor, *Associative Networks*, p. 3–50. Academic Press, New York, 1979.
3. B. C. Bruce. A model for temporal references and its application in a question answering program. *Artificial Intelligence*, 3:1–25, 1972.
4. Z. Cui, A.G. Cohn, and D.A. Randell. Qualitative simulation based on a logic of space and time. In *Proc. of QR92*, Heriott Watt University, 1992.
5. Dennis DeCoste. Dynamic across-time measurement interpretation. *Artificial Intelligence*, 51:273–341, 1991.
6. K. D. Forbus. Qualitative process theory. *Artificial Intelligence*, p. 85–168, 1984.
7. K. D. Forbus. Interpreting measurements of physical systems. In *Proceedings of AAAI-86*, p. 113–117, 1986.
8. Christian Freksa. Temporal reasoning based on semi-intervals. *Artificial Intelligence*, 54:199–227, 1992.
9. T. Fuhr, F. Kummert, S. Posch, and G. Sagerer. An approach for qualitatively predicting relations from relations. In Erik Sandewall and Carl Gustav Jansson, editors, *Proc. of the Scandinavian Conference on Artificial Intelligence*, p. 38–49, Amsterdam, 1993. IOS Press.
10. H. W. Guesgen. Spatial reasoning based on allen's temporal logic. Techn. Report TR-89-049, Int. Computer Science Institute, Berkeley, August 1989.
11. Daniel Hernandez. *Qualitative Representation of Spatial Knowledge*. PhD thesis, Technical University of Munich, 1992.
12. Henry A. Kautz and Peter B. Ladkin. Intergrating metric and qualitative temporal reasoning. In *Proceedings of AAAI-91*, p. 241–246, 1991.
13. F. Kummert, E. Littmann, A. Meyering, S. Posch, H. Ritter, and G. Sagerer. A Hybrid Approach to Signal Interpretation Using Neural and Semantic Networks. In *Mustererkennung 93, 15. DAGM-Symposium*, p. 245–252. Springer, Berlin, 1993.
14. F. Kummert, H. Niemann, R. Prechtel, and G. Sagerer. Control and Explanation in a Signal Understanding Environment. *Signal Processing, special issue on 'Intelligent Systems for Signal and Image Understanding'*, 32:111–145, 1993.
15. Alan K. Mackworth and Eugene C. Freuder. The complexity of some polynomial network consistency algorithms for constraint satisfaction problems. *Artificial Intelligence*, 25:65–74, 1985.
16. H. Niemann, G. Sagerer, S. Schröder, and F. Kummert. ERNEST: A Semantic Network System for Pattern Understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(9):883–905, 1990.
17. G. Sagerer. Automatic interpretation of medical image sequences. *Pattern Recognition Letters*, 8:87–102, 1988. Special Issue on Expert Systems in Medical Imaging, Elsevier Science Publisher, Amsterdam.
18. G. Sagerer. *Automatisches Verstehen gesprochener Sprache*, volume 74 of *Reihe Informatik*. Bibliographisches Institut, Mannheim, 1990.
19. Daniel S. Weld and Johan de Kleer, editors. *Readings in Qualitative Reasoning about Physical Systems*. Morgan Kaufmann, San Mateo, California, 1990.