Qualitative Motion Representation in Egocentric and Allocentric Frames of Reference

(This paper has appeared in: Christian Freksa and David M. Mark (eds.): Spatial Information Theory. Cognitive and Computational Foundations of Geographic Information Science (COSIT 99). Lecture Notes in Computer Science No. 1661. Springer Berlin, Heidelberg, New York, 1999, Pages 461-467)

Alexandra Musto[1], Klaus Stein[1], Kerstin Schill[2], Andreas Eisenkolb[2], and Wilfried Brauer[1,*]

[1] Institut für Informatik,
Technische Universität München,
D-80290 München, Germany
{musto,steink,brauer}@in.tum.de
[2] Institut für medizinische Psychologie,
Ludwig-Maximilians-Universität München,
D-80336 München, Germany
{kerstin,amadeus}@imp.med.uni-muenchen.de

Abstract. In qualitative motion representation, frames of reference play an important role as well in measuring of the motion data as in representation and application of algorithms. This paper discusses motion representation in egocentric and allocentric frames of reference and begins with some general considerations on motion representation through qualitative distances and directions that apply to both technical and biological systems. An approach that involves incremental numeric generalization of a numerically represented motion track and subsequent transformation in a qualitative representation has advantages for technical systems, though. Last, algorithms for generalizing qualitatively represented motion tracks in egocentric and allocentric frames of reference are presented.

Keywords. Spatial Reasoning, Qualitative Reasoning, Representation of Spatio-temporal Knowledge, Spatial Reference Frames.

1 Introduction

The understanding of motion perception and representation is not only essential for the comprehension of the human visual system, but also important for the

* We thank the reviewers for their helpful comments.

development of artificial systems that move or deal in some way with moving objects. When these systems interact with humans, this understanding is especially important, since a user interface should be user friendly and therefore reflect the peculiarities of human spatial cognition instead of forcing the user to communicate with the system in an unnatural manner. This is where considerations on qualitative representations of motion are necessary, since humans mostly think in qualitative categories like left/right, slow/fast or near/far (see, e. g., [11]) and not in quantitative ones like 139° or 78cm. General considerations on qualitative motion representation, though, can be seen as basis for the understanding of a possibly qualitative motion representation in biological systems as well as prerequisite for its application in technical systems.

In the following, we deal with motion of objects in 2d space, whereby we ignore the expansion of the object. First, we take a closer look at the relevancy of reference frames in qualitative motion representation and develop formalisms for measuring a course of motion in the egocentric and the allocentric frame of reference, show how this can yield the same representations, and point out difficulties. Then, in the following section, we deal with the task of generalizing motion tracks, what means to simplify them so that the fine structure is suppressed and only the coarse structure, containing the information on major directional changes and the overall shape of the course of motion is left. We give algorithms for generalizing numeric and qualitative motion tracks in the egocentric and allocentric frame of reference.

2 Qualitative Motion Representation

2.1 Frames of Reference in Measurement and Representation

When talking about spatial representations, we have also to talk about reference frames. A discussion on the different notions of reference frames can be found in [4, 5]. In this paper, we use the notions of egocentric and allocentric reference frames as defined in [4].

To make things a little bit more more complicated, we want to point out that in motion representation we have to distinguish two different levels at which different reference frames may be used: The lower level is the level of *measurement* of the motion event, the level above is the level of *representation*. Imagine you were a participant in the rallye Granada-Dakar and have measured your course of motion of the day's stage through a GPS-Track, that is in an allocentric frame of reference. In analyzing it you could easily transform this track into a representation in the egocentric frame of reference, and make a statement like "In this village we made a mistake and turned to the right instead of turning to the left". In this case, a motion event that was measured in the allocentric frame of reference is represented in the egocentric one. So, in motion representation, it is not only important which frame of reference is used, but also on which level it is used. Measurement and representation form two independent layers in the task of motion representation, where appropriate frames of reference can be chosen:

We can measure motion data from an egocentric point of view without an allocentric, fixed coordinate system, e.g. if we count steps and memorize the angles when turning. Then the frame of reference in measurement is egocentric. Then we can represent this data also in an egocentric frame of reference; the frame of reference in representation is then egocentric, too. Or we can try to map the egocentrically measured data into a global coordinate system and represent the course of motion in an allocentric frame of reference in representation.

On the other hand, we can measure motion data from an allocentric point of view with a fixed coordinate system, e.g. when we store the GPS-Track of a certain route we traveled. We then are able to transform the measured data into an egocentric frame of reference, e.g. for a route description. Then, the frame of reference in measurement is allocentric, and the frame of reference in representation is egocentric.

2.2 Some General Considerations on Motion Representation through Qualitative Distances and Directions

For many applications and especially in the case of locomotion, it is not necessary to model positional information, but only positional change. In biological systems, the availability of positional information seems at least doubtful. Therefore it suffices to represent only distance and direction of a motion event. To this end we use qualitative categories like "left", "right", "far", "close", etc. Then, a course of motion can be represented by a sequence of qualitative motion vectors (QMVs), i.e. vectors that describe the motion of an object from position $n-1$, measured at measurement point $n-1$, to position n, measured at measurement point n, where the vector components are some qualitative descriptions of, e.g., direction, distance, and speed of the object when moving from point $n-1$ to point n. Assuming a fixed scan rate, speed is a derived measure and can simply be computed from the distance covered in one scan cycle.

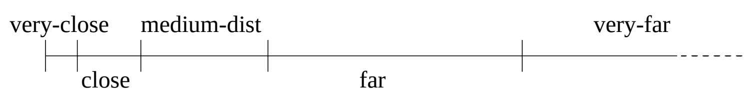

Fig. 1. A discretization of the distance domain

To represent a course of motion through qualitative directions and distances, we have to discretize space somehow into areas to which these categories apply. Figures 1 and 2 are examples of possible discretizations.

Allocentric Frame of Reference in Measurement In [8], a qualitative representation of the course of motion in an allocentric frame of reference in measurement and, first, allocentric

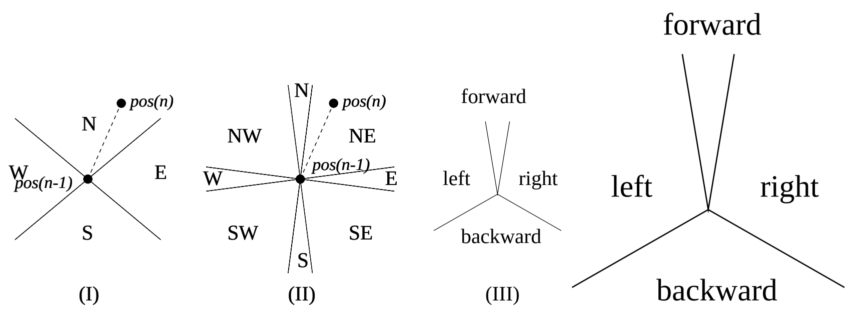

Fig. 2. Left: Discretizations of direction (allocentric) – Right: Asymmetric direction intervals (egocentric)

frame of reference in representation by means of qualitative motions vectors (QMV's) is described. The formalization is done relatively straight forward:

A course of motion is measured with a fixed scan rate and qualitatively represented by the components distance and direction. Speed can be derived from the distance the moving object covered in a single scan cycle. Space is discretized sharply in the domain of distance and direction like in Figure 1 and 2(I), following the suggestions of [1]. So, a course of motion is represented as sequence of qualitative motion vectors, e.g.

`<close east>`^{5} `<close north>`^{2} `<close west>`^{3} `<close south>`^{1}
`<medium-dist south>`^{1} `<medium-dist east>`^{1}.

The indices indicate for how many scan cycles no change in distance and direction occured.[1]

If the course of motion is measured in an allocentric frame of reference in measurement like here, switches between allocentric and egocentric frame of reference in representation in the domain of orientation are possible without loss of information. A representation in the egocentric frame of reference in representation of the above QMV sequence reads:

`<close forward>`^{5} `<close left>`^{2} `<close left>`^{3} `<close left>`^{1}
`<medium-dist forward>`^{1} `<medium-dist left>`^{1}.

Unfortunately, this approach doesn't work when using an egocentric frame of reference in measurement. Since there is no external, fixed, and relatively unchanging frame of reference, the direction gridlock has to be newly aligned in each scan cycle, depending on the new intrinsic orientation of the moving person or robot. Therefore, small changes in the direction in each scan

[1] Mapping of distance combined with the counter-information (e.g. 5 times "close" yields "medium-dist") and speed in qualitative intervals yields a representation like this: **`<medium-dist east slow>` `<close north slow>` `<close west slow>` `<close south slow>` `<medium-dist south medium-vel>` `<medium-dist east medium-vel>`** (vel = velocity).

cycle might be never noticed, but may accumulate to a rather big change in many scan cycles. E.g., if we measure direction under these circumstances with a coarse discretization like in Figure 2 (I), and we make only small changes in direction in each scan cycle (e.g. 20°), we would never notice, even if we had made a whole loop in the end (cf. Figure 3 (Left)). The only changes in direction we would notice would be very sharp ones in one single scan cycle. This leads to the dissatisfactory situation that the representation of the spatial path of a course of motion at a given scan rate depends greatly on the speed of the moving object.

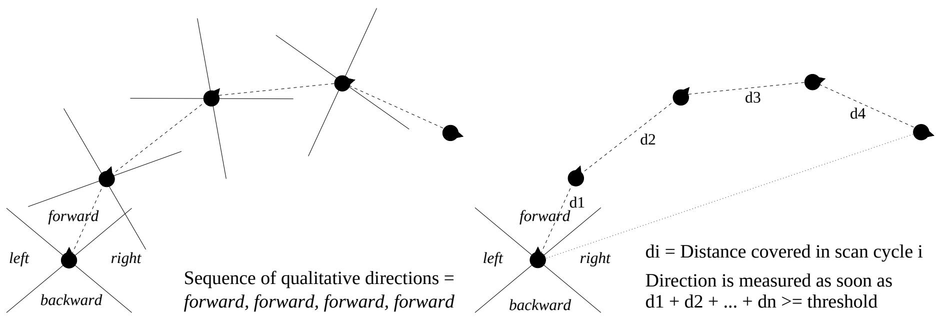

Fig. 3. Left: Small changes in direction may be problematic – Right: Solution

Egocentric Frame of Reference in Measurement So, for the case of an egocentric frame of reference in measurement we have to take into account these special circumstances when we want to create an adequate qualitative representation.

We have several possibilities how to avoid the above problem, that is, e.g.:

1. Mapping of egocentrically measured data into an allocentric coordinate system, and then applying the discretization procedure. Then we virtually use an allocentric frame of reference in measurement and all algorithms in [8] could be used. The disadvantage is that all measured egocentric data has to be converted into some allocentric data in a totally arbitrary coordinate system to apply an allocentric frame of reference in measurement, from where we get an allocentric representation in the frame of reference in representation, which can then again be converted back into egocentric represented data in the frame of reference in representation.
2. Figure 2 (Right) shows a solution of the problem using special acceptance areas. For this areas not being symmetric, the application of some algorithms to the representation would be more difficult. In general, unequal or even unsymmetric discretizations of space are no problem when measuring motion

data from an allocentric point of view, but may cause problems in an intrinsic frame of reference because of lessened rotation invariance.

3. Accumulating changes in direction until a new direction region is reached. This has the disadvantage that a possible measurement error may greatly accumulate which would normally be filtered away by the qualitative representation but comes so again into account.
4. Measuring changes in direction not in every scan cycle, but only after a certain distance was covered (see Figure 3 (Right)). This is the reverse paradigm to [8], where time is fixed and distance is variable in the measurement. Here, we fix the distance and therefore have to measure the time needed to cover the fixed distance in order to derive speed. This makes sure that the resulting representation doesn't depend on the speed of the motion.

All possibilities guarantee reasonable consistency of the spatial representation independent of the speed of the moving object. In the latter, we stick to the last one for symmetry with the presented approach in the allocentric frame of reference in measurement.

egoQMV's Following [8], we can then represent locomotion as a sequence of egoQMV's, measured and represented in the egocentric frame of reference. At first, an egoQMV consists of the components

- Direction D, e.g. {`forward, backward, left, right`}.
- Number of time cycles t needed to cover the fixed distance.

A counter indicates for how many measurement cycles no change in direction and speed[2] has occured. A course of motion is then represented as a sequence of egoQMV's, for example:

`<forward 6>`^{5} `<left 6>`^{2} `<left 6>`^{3} `<left 6>`^{1} `<forward 4>`^{6}
`<right 4>`^{6}.

We can then resolve the time into speed and so represent speed somehow more qualitatively, too. That is, the speed information is directly derived from t: the moving object needed 6 time cycles to cover the measurement distance, which maps to a `slow` motion:

`<forward slow>`^{5} `<left slow>`^{2} `<left slow>`^{3} `<left slow>`^{1}
`<forward medium-vel>`^{6} `<right medium-vel>`^{6}.

At last, distances can also be mapped into some distance intervals. The distance information is computed from the counter combined with the measurement distance: in the first vector, the moving object went 5 times the measurement distance forward, which yields a `medium-dist`:

[2] Since we measure after a fixed distance, t is equivalent to speed: the larger the number, the slower the robot is moving.

```
<medium-dist forward slow> <close left slow>
<close left slow> <close left slow>
<medium-dist forward medium-vel>
<medium-dist right medium-vel>.
```

This representation can be transformed into an allocentric frame of reference in representation if the allocentric heading of the object when starting the motion is known. Actually, if the allocentric heading at the starting point is assumed to be east, we get the same sequence as in the previous, allocentric, example:

```
<medium-dist east slow> <close north slow>
<close west slow> <close south slow>
<medium-dist south medium-vel> <medium-dist east medium-vel>.
```

It is clear that the same representation of the motion event at the representational level can be obtained regardless of the frame of reference chosen for the measurement level. Nevertheless we have to take into account that we possibly make big mistakes in the transformation from egocentric to allocentric frame of reference, since we rotate the direction grid constantly when mapping the direction of motion into the qualitative interval in the egocentric frame of reference in measurement, whereas in the allocentric frame of reference in measurement this grid is fixed.

2.3 Qualitative Motion Representation in Technical Systems

Whereas mapping distance and direction directly in each scan- or measurement cycle into qualitative intervals like described above may be necessary for some applications and is perhaps performed by biological systems, it has the disadvantage that much information is lost if numeric information is available in the first case, like, e. g., in a moving robot. Then, a better alternative is to memorize the numeric motion data, to generalize it to smooth away irrelevant deviations, and so to create a numeric representation that only contains the few significant changes in direction and speed. Then, the generalized numeric representation can easily be converted into a qualitative one by mapping the changes in direction and speed into qualitative intervals. However, this approach has two disadvantages: on the one hand, a lot of irrelevant data has to be stored, and on the other hand, the qualitative description of the course of motion can only be generated when the motion is finished. Since we are in the happy position to dispose of *incremental* generalization algorithms (see section 3.3 and 3.4), we can avoid these disadvantages and create a qualitative representation of the motion track directly after each segment has been generalized.

3 Generalization

For many purposes in motion representation it is not necessary to store the course of motion with the accuracy it was measured. Imagine, for example, you attend a conference in a foreign city where you stumbled by chance into an ancient church one evening. The next day, you want to give your colleague a route description from your hotel to the church. Certainly, you do not mention every move you made on your way from the hotel to the church, when you did, e. g., some window shopping or strolled over the market, but only the big, relevant directional changes at crossings, etc.

Generalization is a similar process: a motion track is simplified and the fine structure is suppressed. Only the coarse structure, containing the information on the major directional changes and the overall shape of the course of motion is left. Such an operation can be useful when processing motion information with a limited memory capacity, e.g. in biological system, but also in technical system. For example, in a system navigating on the basis of its own recorded locomotion tracks, like the Bremen Autonomous Wheelchair as described in [7], a track description intended to guide later movements is insofar similar to the above example of a route description that mostly not the most accurate description is best. E.g., to find out which corridor in the building the wheelchair went it is not important if it did this next the left or right wall or if it had to maneuver a few times to be able to do the turn. This information would obstruct, since next time the robot might take a slightly different track and so perhaps needs not to do the same maneuvers.

Analytical generalization algorithms as used by, e.g., cartographers [2, 3], are mostly not suitable for our purpose, since the computational effort of computing a smooth curve is not necessary with our discrete representation, but so considerable, that it would perhaps not be possible to generalize "on the fly" in a real time environment. Other Algorithms like the one described in [9] need the completed motion path and are therefore also not suitable for processing on the fly.

In the following, we describe means to calculate with QMV's and two generalization algorithms that can be applied to courses of motion given in numeric coordinates and to allocentric and egocentric QMV sequences.

3.1 Calculation on Motion Sequences

A generalization algorithm is a function taking a motion track (a vector sequence) as input and returning another (somehow "simpler") motion track. Therefore, since we should be able to do calculations on motion tracks, we need an algebra.

Numeric representation A course of motion given in numeric representation simply is a vector sequence in ordinary two-dimensional space. All the well known methods of linear algebra can be used.

Allocentric QMV representation By choosing a discretization of the distance domain that allows to express larger distances by smaller ones (e. g. `close` $= 3 \times$ `very-close`),

$$e^1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \texttt{<very-close east>},$$
$$e^2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \texttt{<very-close south>}$$

form a two dimensional space similar to the numeric one above, where any QMV can be represented. Not every member of this QMV space can be represented as single QMV (e. g. $\begin{pmatrix} 7 \\ -5 \end{pmatrix} = (\texttt{<very-close east>}^7, \texttt{very-close north>}^5)$), so at end of calculation these vectors must be mapped back to QMV (e. g. $\begin{pmatrix} 7 \\ -5 \end{pmatrix} \equiv \texttt{<very-close northeast>}^9$, or, in case of only four directions, $\texttt{<very-close east>}^9$) the same way numeric vectors are mapped to QMV.[3]

Egocentric QMV representation As mentioned before, any egocentric QMV sequence can always be mapped to an allocentric frame of reference and the algebra described above can be used. A QMV sequence recorded in an egocentric frame of reference in measurement normally contains systematic distortions accumulating along the sequence. So here it may be interesting to give up the universality of representation in a two dimensional space and to restrict to local combination of successive QMV.

3.2 Local Egocentric Calculation

The main goal here is to have some means to combine two successive egoQMV's. The domain of Directions forms a group with `forward` as neutral element being inverse to itself, `backward` being inverse to itself and `left` and `right` being inverse to each other. But combination of directions alone is useless, so mainly the interplay of direction and distance in egoQMV combination is of interest.

In [1], a solution to this problem is suggested. It is based on interval arithmetic and yields correct, but vague results, namely a disjunction of possible qualitative intervals. Although this is a correct and working solution, it does not fit to our approach of qualitatively representing motion and applying algorithms to this representation: Since we deal with possibly thousands of egoQMV's we would in the end get a result that is correct, but useless, since it would simply be a disjunction of all possibilities.

So, we rather accept that we make a certain error in calculation but get no disjunction of intervals as result.

How two egoQMV's are combined depends on distance and direction. As in allocentric representation we need a discretization of distance that allows to express

[3] This QMV algebra is described in detail in [8].

larger distances by smaller ones. We can distinguish several cases, but since direction is measured and represented egocentrically, only the direction of the second egoQMV matters (the description below is done for four directions. Having 8 directions the idea is mainly the same, it just gets much more complicated):

1. <dist1 dir1 speed1> + <dist2 *forward* speed2> =
   <dist1+dist2 dir1 average-speed>
2. <dist1 dir1 speed1> + <dist2 *backward* speed2> =
   <|dist1-dist2| erg-dir erg-speed>.
   If dist1 - dist2 > 0, erg-dir = dist1; else erg-dir = -dist1.
   erg-speed = |dist1-dist2| divided by the sum of the total time.
3. The direction of the second vector is left or right. If we deal only with four directions, we can't compute an intermediate direction. So, we have to use a rule of absorption, that is, the bigger vector somehow absorbs the smaller one regarding the direction. This is plausible when we imagine the direction grid. Since it is newly aligned in each scan cycle, the reference direction points directly forward in the middle of the interval.

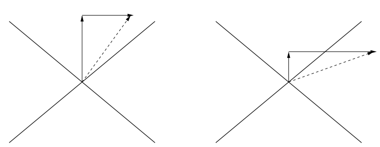

Fig. 4. Absorption rule

Since the resulting direction must be represented in relation to the direction before, we have here to take a look the possibilities. In the following is dir $\in$ {left, right}.
- <dist1 dir speed1> + <dist2 dir speed2> =
  <dist2 backward speed2>, if dist2 > dist1;
  <dist1 dir speed1> otherwise.
- <dist1 dir speed1> + <dist2 -dir speed2> =
  <dist2 forward speed2>, if dist2 > dist1;
  <dist1 dir speed1> otherwise.
- <dist1 forward speed1> + <dist2 dir speed2> =
  <dist1 forward speed1>, if dist1 > dist2;
  <dist2 dir speed2> otherwise.
- <dist1 backward speed1> + <dist2 dir speed2> =
  <dist2 -dir speed2> = if dist2 > dist1;
  <dist1 backward speed1> otherwise.

The absorption rule certainly causes a systematic shift of the rest of the sequence. In case of 8 directions the error is smaller for e. g. forward and left can be combined to halfleft, but even then absorption has to be done (e. g.

halfright + right). So it could be useful not just to combine any two successive vectors, but to start with successive vectors of same (forward) or opposite (backward) direction, in case of 8 directions followed by successive vectors where directions can be combined to a new one, and last all the combinations where the absorption rule is needed.

In case 2 and 3 of the QMV additions described above it is possible that the resulting vector has a direction that differs in the allocentric frame of reference in representation from that of the second vector participating in the operation. We have to take this into account when we perform these operations in an egoQMV sequence, where the direction of the following egoQMV is relative to that of the second vector in the operation. Therefore, we need to compute the new relative direction for the following vector.

Consider three subsequent vectors A = <dist1 dir1 speed1>, B = <dist2 dir2 speed2> and C = <dist3 dir3 speed3> in an egoQMV sequence, where vectors A and B are combined. If the resulting vector has the same direction as B, nothing has to be done with C. A new egocentric direction has to be computed for C, if the combination of vectors A and B yields another direction than that of B as result.

This can be the case if

1. the subtraction of a backward-vector reverts the egocentric direction (A + B = <|dist1-dist2| -dir1 erg-speed>): Then the direction of C has to be inverted too and C becomes <dist3 -dir3 speed3>.
2. the absorption rule yields A as result. Then the directions of B and C have to be combined to realign C. Qualitative directions being a group, this works as follows (dir $\in$ {left, right}):
   - dir $\circ$ -dir = forward
   - dir $\circ$ dir = backward
   - dir $\circ$ forward = dir
   - dir $\circ$ backward = -dir

3.3 Incremental ε-Generalization

The basic idea of generalizing a given track T is to find a simpler track U representing the general shape of T—i. e. the important global information—and suppressing small zigzags and deviations. A simple approach to generalize T is to create a polygon track U differing less than a distance ε from T (ε requirement).

An incremental algorithm based on this idea is illustrated in Figure 5: starting from A, the algorithm follows the polygon track, tries to build a straight line segment from the actual starting point to the current point and tests whether each point between the starting point and the current point is within the ε-surrounding of the generalized line segment. In Figure 5 II, the distance of the points B and C to [AD] is tested to be less than ε. So, the line [AD] qualifies as generalization of the polygon track between A and D, and the algorithm tries

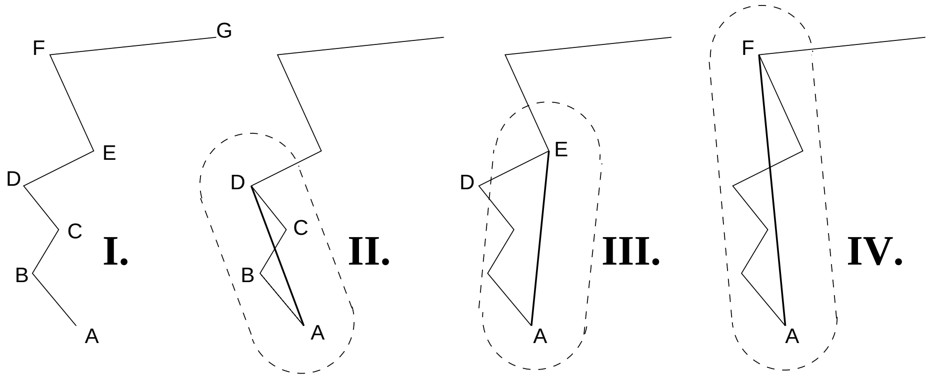

Fig. 5. Steps during generalization

to proceed by taking the next point E as new end point for the generalized line segment. In this case, the point D is outside the ε-surrounding of the generalized line segment (cf. Figure 5 III). Now the algorithm tests whether there could be a further reference line closer to this point when the polygon track proceeds. This is the case with line [AF]. If no further reference line could fulfill the ε requirement, or if the input stream is closed, the last correct reference line segment is taken and its endpoint is used as the end of the first part of the generalized track, and simultaneously this point is the start for the next iteration.

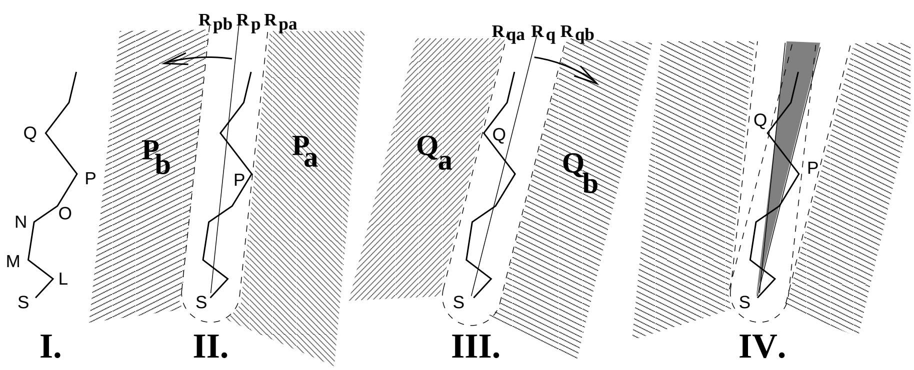

Fig. 6. Efficient checking of the ε-requirement

Figure 6 visualizes the algorithm that tests whether the construction of a further reference line fulfilling the ε-requirement is possible: the first end point of the reference line segment is the starting point S and the line can rotate around this point. Any point further than ε away from S limits the angle of rotation for any longer reference line segment starting with S: for example, since S and L are not farther than ε away from each other, any reference line would fulfill the requirement. The addition of point N to the track limits the possible directions of rotation: the line could not go backwards from S, since S and N are farther than ε away from each other.

Continuing with the example, in Figure 6 II, P prevents any further rotation of the reference line shown to the left, in Figure 6 III Q prevents a rotation to the right. Therefore, for any point in $\mathbf{P}_b$ or $\mathbf{Q}_b$ it is impossible to find a reference line ($\mathbf{R}_p$, $\mathbf{R}_{pa}$, $\mathbf{R}_{pb}$, $\mathbf{R}_q$, $\mathbf{R}_{qa}$, $\mathbf{R}_{qb}$ are easy to compute).

The algorithm works as follows:

1. The point X_1 and the second point X_2 are the starting point and end point of the first reference line segment S_1.
2. Then it is checked whether any new point X_n is either in $\mathbf{P}_b$ or in $\mathbf{Q}_b$.
3. If this is not the case, it is tested whether X_n is in $\mathbf{P}_a$ or in $\mathbf{Q}_a$. Then X_n is used as the new border point and $\mathbf{R}_p$, $\mathbf{R}_{pa}$, $\mathbf{R}_{pb}$ (or $\mathbf{R}_q$, $\mathbf{R}_{qa}$, $\mathbf{R}_{qb}$) are recomputed.
4. Otherwise, the end of a segment has been passed, and the end point X_k ($k < n$) of the last correct reference line can be used as end point of the segment starting with X_1. Then, the algorithm continues with (1) to calculate the next line segment S_2, etc.

The algorithm generates a generalized vector sequence, where all corners of the generalized track are also corners of the original track (so the result can *never* be more complex than the original), and where no point of the original track has a distance greater than ε to the generalized one.

A main advantage is, that due to the incremental approach the generalization can be calculated during measurement. In real-time environments it is even possible to give a coarse prediction of the direction of the actually computed vector.

Due to this linear computation, the average complexity is $O(n)$, in worst case (lot of backtracking) $O(n^2)$, which is nearly never reached.

As described in section 3.1 the algorithm can be used on numeric data and on QMV sequences. It was successfully applied in robot navigation by locomotion data, showing good results [7]. The robot has to navigate in a building with corridors and rooms and so the width of a corridor gives a good value for ε and the generalization therefore suppresses the small movements of the robot in the corridor, showing the main movement along the corridors and into the rooms. It is also useful for creation of a qualitative description of a course of motion, e.g. for user interaction in a semi-autonomous system. By generalizing the course of motion incrementally, a qualitative representation of the simplified track can be created on the fly.

We don't deal with speed information in this algorithm at the moment, but it is simple to calculate the average speed of each generalized segment. However, big changes of speed are, at the moment, no criterion for generalization (like an additional or combined ε-criterion). This will be a topic for future work.

3.4 Incremental Σ-Generalization

The main idea of the generalization algorithm is to take a certain step size, a distance, as parameter, and to sum up all vectors falling into one step. This smoothes the curve (It is an extension of the idea of measuring after a certain covered distance described before). To get no artificial corners that don't appear at all in the original course of motion, we take the step size as approximate bound, only "cutting" at "real" corners (that is, where direction of motion changes).

The algorithm itself is very simple. With a given generalization distance σ:

1. Successive vectors of the same direction are added and count as one (larger) vector (This is not very important at numeric motion data but a frequent case in processing QMV sequences), vectors of length 0 are deleted.
2. The length of all vectors is summed up until σ is reached (trying to get as close as possible to σ):
$$\begin{aligned} \sigma_l &= \textstyle\sum_{i=1}^{n} v_i \leq \sigma \\ \sigma_u &= \textstyle\sum_{i=1}^{n+1} v_i > \sigma \end{aligned}$$
If $|\sigma - \sigma_l| < |\sigma - \sigma_u|$ the first n vectors are summarized, otherwise $n+1$.
3. The generalized vector is the sum of the n (or $n+1$) vectors above.
4. Especially when proceeding QMV sequences, successive vectors with same (qualitative) direction in the generalized track should be combined to one (larger) vector.

Certainly all this can be done in one step.

The algorithm has a complexity of $O(n)$ and the number of vectors the generalized track consists of is limited by the value of σ and the length of the original track (It is easy to see that it approximately consists of (length of original track)/σ vectors, with the maximum number being 1.5 times of this).

The algorithm reduces complexity by simply summing up successive vectors, therefore it is not really sensitive for big corners, but it is obvious that small movements (that is much smaller than σ) are suppressed. Since we have to take a look at a whole subsequence of the QMV sequence to simplify it, we need some kind of spatio-temporal storage. In [10], a model for such a memory in human visual cognition is suggested.

One big advantage is that the algorithm can be used with local egocentric calculation: the distances are given directly in the QMV and the calculation of the generalized vector (step 3) can be done as described in section 3.2, so generalization can be done directly on QMV sequences with no need to map and remap to and from a two dimensional vector space. Certainly it can also be applied to numeric vector sequences and allocentric QMV sequences.

3.5 Properties of Generalization

It is easy to see that both algorithms suppress small zigzags in motion tracks but keep the main shape:

In numeric motion data the corners of the generalized track are also corners of the original track, so at each corner the moving object (or subject) following the generalized track is guaranteed to reach the same place as it would by following the original track, but at a different angle. This has to be taken into account by using route descriptions like: "...there you see a lemon tree at your right", for by following the generalized track it could appear in front or back instead.

In qualitative motion data the generalized tracks are 'shifted'. This is a result of the mapping process, where different angles (and numeric distances) are represented by the same qualitative directions (and distances).[4]

This is not a real disadvantage, because the shape of the track is represented very well and a QMV sequence is not meant for determinating the exact position of a moving object anyway. The main purpose is the shape of the motion, positional information can be given additionally or incorporated by landmarks, as described in [6].

4 Conclusion and Outlook

In this paper, we stated the necessity to introduce the distinction between frame of reference in measurement and frame of reference in representation when qualitatively representing the course of motion. A general discussion of problems of motion representation through qualitative distances and directions builds a basis for the understanding of a possibly qualitative motion representation in biological systems as well as prerequisite for its application in technical systems, where qualitative representations are especially interesting when designing user interfaces. Nevertheless, for some purely technical applications, qualitative motion representation may have the disadvantage of being too imprecise. Therefore, in technical systems with user interaction, where sufficient exact numeric information is available, qualitative and quantitative representations should coexist. Incremental numeric generalization algorithms allow for quick and efficient abstraction from the fine structure of a course of motion and easy transformation in qualitative categories. However, if numeric information is not available, generalization is nonetheless possible in egocentric as well as in allocentric frames of reference.

The incremental ε-generalization algorithm was applied successfully in robot navigation in closed buildings, where it was used to simplify stored locomotion tracks and to navigate by comparing actual motion tracks with the stored ones [7]. Since the abovementioned robot is a semi-autonomous wheelchair, where

[4] This shift therefore occours at any transformation from numeric to qualitative representation

a simple user interface is necessary, a qualitative representation of locomotion might be additionally used in the future for simplifying user interaction.

References

1. Eliseo Clementini, Paolino Di Felice, and Daniel Hernández. Qualitative representation of positional information. *Artificial Intelligence*, 95(2):317–356, 1997.
2. Douglas and Peucker. Algorithms for reduction of the number of points required to represent a digitized line or its caricature. *The Canadian Cartographer*, 10/2, 1973.
3. Emmanuel Fritsch and Jean Philippe Lagrange. Spectral representations of linear features for generalisation. In Andrew U. Frank and Werner Kuhn, editors, *Spatial Information Theory. A Theoretical Basis for GIS. European Conference, COSIT'95*, volume 988 of *Lecture Notes in Computer Science*, pages 157–171, Berlin, Heidelberg, New York, September 1995. Springer.
4. Roberta L. Klatzky. Allocentric and egocentric spatial representations: Definitions, distinctions and interconnections. In C. Freksa, C. Habel, and K. Wender, editors, *Spatial Cognition. An Interdisciplinary Approach to Representing and Processing Spatial Knowledge*, volume 1404 of *Lecture Notes in Artificial Intelligence.* Springer, 1998.
5. S. C. Levinson. Frames of reference and Molyneux's question. In P. Bloom, M. A. Peterson, and L. Nadel, editors, *Language and Space*, pages 109–169. MIT Press, Cambridge, MA, 1996.
6. Alexandra Musto. On spatial reference frames in qualitative motion representation. Forschungsberichte Künstliche Intelligenz FKI-230-99, Institut für Informatik, Technische Universität München, 1999. Submitted to "Spatial Cognition and Computation", Kluwer Academic Publishers.
7. Alexandra Musto, Klaus Stein, Andreas Eisenkolb, and Thomas Röfer. Qualitative and quantitative representations of locomotion and their application in robot navigation. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI-99)*, pages 1067–1073, San Francisco, CA, August 1999. Morgan Kaufman Publishers, Inc.
8. Alexandra Musto, Klaus Stein, Andreas Eisenkolb, Kerstin Schill, and Wilfried Brauer. Generalization, segmentation and classification of qualitative motion data. In Henri Prade, editor, *Proceedings of the 13th European Conference on Artificial Intelligence (ECAI-98)*, pages 180–184. John Wiley & Sons, 1998.
9. Jonas Persson and Erland Jungert. Generation of multi-resolution maps from run-length-encoded data. *International Journal of Geographical Information Systems*, 6(6):497–510, 1992.
10. Kerstin Schill and Christoph Zetzsche. A model of visual spatio-temporal memory: The icon revisited. *Psychological Research*, 57:88–102, 1995.
11. Barbara Tversky. Cognitive maps, cognitive collages, and spatial mental models. In Andrew U. Frank and Irene Campari, editors, *Spatial Information Theory. A Theoretical Basis for GIS. European Conference, COSIT'93*, volume 716 of *Lecture Notes in Computer Science*, pages 14–24, Berlin, Heidelberg, New York, September 1993. Springer.