Action Vectors and Its Application to Interactive Dialogue Systems

Junko Araki[†] **Takashi Ninomiya**[‡*] **Takaki Makino**[¶] **Jun'ichi Tsujii**[‡*]

[†] Interfaculty Initiative in Information Studies, University of Tokyo
[‡] CREST, Japan Science and Technology Corporation
[*] Department of Computer Science, University of Tokyo
[¶] Department of Complexity Science and Engineering, University of Tokyo
address: Hongo 7-3-1, Bunkyo-ku, 113-0033, Tokyo phone: +81-3-5803-1697 FAX: +81-3-5802-8872
e-mail: {jun3, ninomi, mak, tsujii}@is.s.u-tokyo.ac.jp

keywords: navigation system, route description, perspective system, human spatial cognition

Abstract

In navigation tasks, the addresser often gives ambiguous route instructions. This paper describes the factor of ambiguity in the route instructions, and discusses the way of developing an interactive dialogue system. We introduce *action vectors*, which are points where the addressee took some action according to the given instructions. In our proposal, ambiguity in interpreting route instructions is explained by indeterminacy in the selection of an action vector. Action vectors enable us to provide a more reliable dialogue system that can interact with users more naturally.

1 Introduction

In navigation tasks, an addresser describes instructions such as 'Turn left' or 'On the right' and disambiguates that kind of instructions by regarding the addressee's current position as a reference, e.g., 'Turn left at *your current position*' (Simmons and Koenig, 1995; Burgard et al., 1998). However, in case that an addressee is given all descriptive instructions in advance (Figure 1) and actually travels around an unfamiliar world (Figure 2), he/she may use his/her former positions as well as his/her current position as the reference of the instructions and interpret 'Turn right' as 'Turn right at *the crossing where you turned a little while ago.*' Unfortunately, this kind of approach has not been well taken in the current studies of the navigation system.

We propose an explanation of ambiguity in interpreting route instructions regarding *an action vector* as a new candidate of a reference of the instructions. The action vectors are the addressee's former positions on the route, where he/she changed his/her behaviors according to the given instructions. The structure of the action vector is useful to make more reliable interactive systems in navigation, where the instruction ambiguity is solved through exchanging dialogue between the user and the system.

> *You can see a crossing straight ahead and turn left there. On the left side there is a post office. Get me a stamp there. then on the right, you can see a flower shop. Don't forget to buy a bouquet of roses for our mother's birthday present. And go straight to get to the cafe. I'm waiting there.*

Figure 1: Description 1: Instructions to get to the destination

2 Ambiguity of Instructions

Consider the world in Figure 2 again. When the addressee follows the instruction *'On the right, you can see a flower shop,'* there are three interpretations depending on the reference of *'On the right.'*

1. On the right at *the doorway from inside of the post office.* (Flower shop 1)
2. On the right at *the doorway from outside of the post office.* (Flower shop 2)
3. On the right at *the crossing where he turned left a little while ago.* (Flower shop 3)

In addition, there are three interpretations for 'go straight' for each of the interpretations above. As a result, Description 1 has nine interpretations.

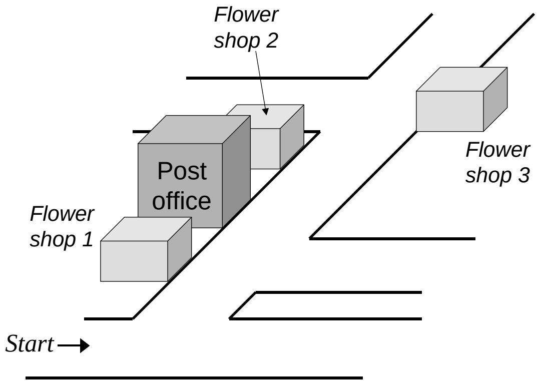

Figure 2: The world where an addressee travels according to Description 1.

d_1	*you can see a crossing ... turn left there*
d_2	*on the left side, you can see a post office ... get me a stamp there*
d_3	*on the right, you can see a flower shop ... buy a bouquet*
d_4	*go straight to get to ...*

Figure 3: Instructions in Description 1

3 Action Vector

Action vectors are specific points on the route, where the addressee turned, stopped or perceived the surroundings to fulfill his/her missions indicated in Figure 3. The definition of action vectors is to be given in this section.

Let $\mathcal{P}$ $(\subset \mathbf{R}^3)$ be the set of all possible vectors of positions on the map, $\mathcal{O}$ $(\subset \mathbf{R}^3)$ be the set of all possible vectors of orientation. Let D be the description, which is the sequence of the instructions $d_1 d_2 d_3 \cdots d_n$ (See Figure 3). Suppose the following situation:

- An addresser gives a description D to an addressee to let him/her do some goal-oriented task. For example, Description 1 in Figure 1 is an example of the description.
- At the time t, an addressee has executed $d_1 d_2 \cdots d_{t-1}$, and is about to execute d_t.

Let $CONT_t$ be all the context that includes the route and any actions which the addressee had taken at the time t, $route(CONT_t)$ be the function that returns the route, and $actvec(CONT_t)$ be the function that returns a set of *action vectors*:

$$actvec(CONT_t) = \{ \quad \begin{array}{l} a_{1,1}, a_{1,2}, \cdots, a_{1,m_1} \\ a_{2,1}, a_{2,2}, \cdots, a_{2,m_2}, \\ \cdots, \\ a_{t,1}, a_{t,2}, \cdots a_{t,m_t} \qquad \} \end{array}$$

where $a_{i,1}, \cdots, a_{i,m_i}$ $(1 \leq i \leq t)$ are action vectors that are generated by executing the instruction d_i, and $a_{i,j}(= \langle p, o \rangle)$ for each i and j $(1 \leq i \leq t,\ 1 \leq j \leq m_i)$ is an action vector such that $p \in \mathcal{P}$, $o \in \mathcal{O}$ and p is the position where the addressee takes an action according to d_i, and o is the orientation for which the addressee is heading according to d_i.

Figure 4 shows an example of $route(CONT_3)$ and $actvec(CONT_3)$ for the route. In the figure, five action vectors are defined: the crossing before and after he turned (a_1 and a_2) corresponding to d_1, the doorway from inside/outside of the post office (a_4 and a_3) for d_2 and the doorway from inside of the flower shop 3 (a_7) for d_3. If the addressee chooses the flower shop 1 at the doorway from outside of the post office (a_3) or flower shop 2 at the doorway from inside of the post office (a_4) according to the instruction 'on the right,' $actvec(CONT_3)$ returns another set of action vectors. Thus, actually we have seven action vectors as shown in Figure 5.

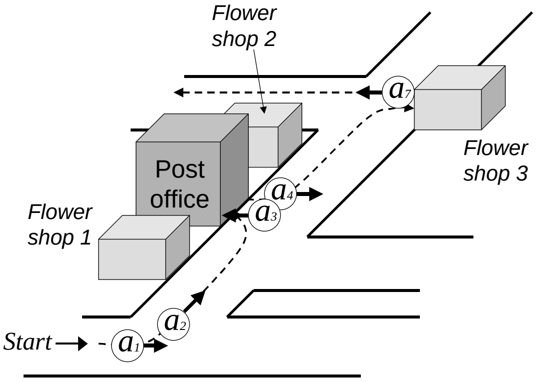

Figure 4: A route in case of choosing a_2 as an optimal action vector for d_3

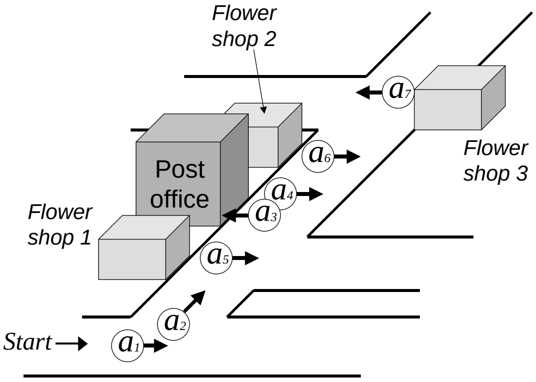

Figure 5: All action vectors given by the Description 1

4 Application

In this section, we discuss an application of the action vector within interactive dialogue of navigation. Although the action vectors are useful to construct an agent system in which a user (addresser) navigates a system (addressee), here we focus on the application in which a system (addresser) navigates a user (addressee).

The instructions without explicit references are often ambiguous. However, it is not practical to give all possible references to all the instructions, since the instructions given along with all references explicitly become long and make the user to be hard to listen to the instructions. To make the navigation more natural, the system should know which instructions are ambiguous.

We can use action vectors as a tool to detect ambiguity in an instruction. For instance, the system should give the instruction 'Turn left' to the user in d_1 (Figure 3), and should not give the instruction with an explicit reference like 'Turn left *at the view from the start position'*. This is because, in this case, the action vector regarded as the reference is only a_1. Even in the case that an implicit reference causes ambiguity in an instruction, to use the notion of the action vector enables the system to disambiguate the instructions through the interactive dialogue.

When the user recognizes ambiguity of the given instruction 'On the right, there is a flower shop' (d_3 in Figure 3), he/she asks back the reference of the instruction 'On the right' to the system like 'Which right is the flower shop located on?' The system has, in advance, memorized the candidates of the action vectors a_1, a_2, a_3, and a_4 as references of the instruction 'On the right' according to the

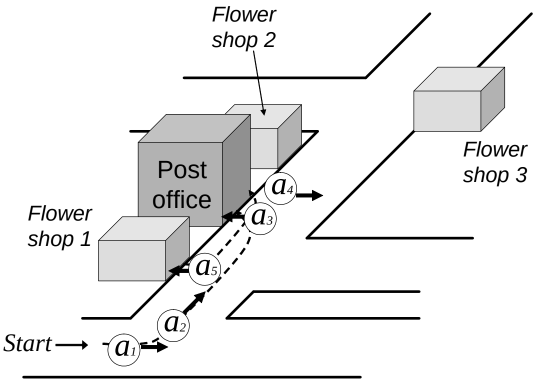

Figure 6: A route in case of choosing a_4 as an optimal action vector for d_3

user's behaviors, who has executed d_1 and d_2. The choice of the action vectors (a_1, a_2, a_3, and a_4) depends on the route plan. Namely, if the system wants the user to head for the flower shop 3, the system selects a_2 for d_3 as an optimal action vector (see Figure 4) and returns the answer converting to a natural language like 'On the right at *the crossing you turned left*.' If the system desires the flower shop 1, the system selects a_4 for d_3 and returns the answer 'On the right of *the doorway from inside of the post office*' (see Figure 6).

In this way, by introducing the action vector to interactive dialogue systems, the instructions without explicit references can be disambiguated, and the systems become more reliable. Furthermore, our new strategy for reliable interactive systems can be applied to the instructions which are dealt with in the former researches on the reference between human spatial cognition and natural language understanding (Tversky and Hermenway, 1984; Herskovits, 1989), the extrinsic (Retz-Schmidt, 1988) and the absolute perspective systems (Levelt, 1986).

5 Conclusion

We gave an explanation of ambiguity of instructions in navigation systems by presenting action vectors as new references. Furthermore, we discussed an application of the action vector to an interactive dialogue system between a user (an addressee) and a system (an addresser) in navigation. By using the notion of the action vector, we proposed a more reliable interactive dialogue system in navigation, which can disambiguate the instructions through interaction between the user and the system.

As our future work, we are developing the computational navigation system using interactive dialogue between the user and the system, by implementing our new concepts of the action vector.

References

W. Burgard, A. Derr, D. Fox, and A. Cremers. 1998. Integrating global position estimation and position tracking for mobile robots: the dynamic markov localization approach. In *proceedings of International Conference on Intelligent Robots and Systems (IROS)*.

A. Herskovits. 1989. *Language and Spatial Cognition. An Interdisciplinary Study of the Prepositions in English*. Cambridge University Press.

W. Levelt. 1986. *Speaking: From intention to articulation*. MIT Press.

G. Retz-Schmidt. 1988. Various views on spatial prepositions. *AI Magazine*, 9(2).

R. Simmons and S. Koenig. 1995. Probabilistic robot navigation in partially observable environments. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI'95)*, pages 1080–1087.

B. Tversky and K. Hermenway. 1984. Objects, parts and categories. *Journal of Experimental Psychology: General*, 113(2):169–191.