Plan Recognition

Marcelo G. Armentano

ISISTAN Research Institute – UNICEN
Tandil, Argentina

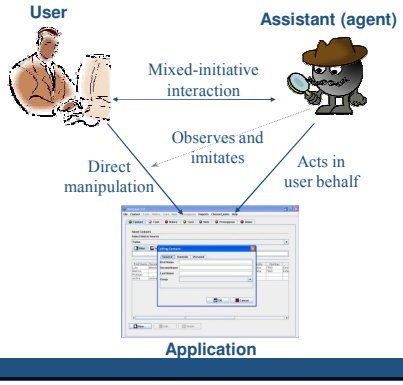
Agenda

- **Problem Overview**
  - Interface Agents
  - Plan Recognition
- **Probabilistic approach**
  - Monitoring user actions
  - Model of the application
  - Model of user intentions
- **Example**
- **Summary**

Interface Agents

- **Assist users in a personalized manner**
- **Learn interests, preferences and needs of the user**
- **Should consider the status of the user's attention and the uncertainty about the user's goals**
  - ability to recognize or predict opportune moments for gaining the user's attention

Interface Agents



Interface Agents

- **Two ways of detecting the user's intention**
  - Asking the user
    - Is a direct way of accessing to his/her intentions
    - We run the risk of disturbing him/her
    - Slows down the interaction
    - Interrupts the user's line of thought
  - Inferring from context
    - Information obtained from user's interaction with the application is on a low level compared to the user's intention

Plan Recognition - Objective

- **Aims at identifying the goal (or intention) of a subject based on the actions he performs in an environment**

Plan Recognition - Objective

- **Aims at identifying the goal (or intention) of the user based on the tasks he performs in a software application**

Some definitions…

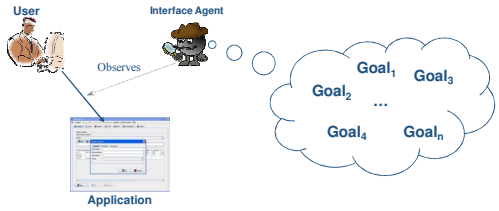
- **Action**
  - Event performed by the user in the GUI of the application.
  - Ex.: Mouse click, key typed, etc
- **Task**
  - Piece of work that the user can perform in the application
  - Ex.: Send an email, add a contact to the address book, enter the topic of a meeting
- **Intention / goal**
  - The desire of the user to do something in the application.
  - Ex.: Arrange a meeting with my friends, Register the birthday of a contact
- **Plan**
  - The set of tasks the user performs to achieve his intention

Plan Recognition – Inputs and outputs

- **Inputs:**
  - a set of goals the agent expects the user to carry out in the domain,
  - a set of plans describing the way in which the user can reach each goal,
  - an action observed by the agent.
- **Output:**
  - foretelling the user's goal, and determining how the observed action contributes to reach it

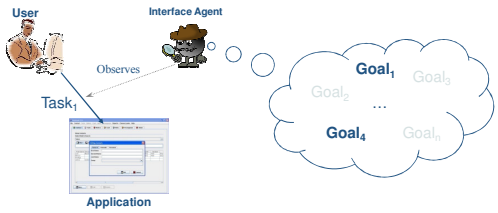
Plan Recognition – Basic Idea

- **narrow the number of possible goals the agent believes the user is pursuing by observing the actions the user performs.**



Plan Recognition – Basic Idea

- **narrow the number of possible goals the agent believes the user is pursuing by observing the actions the user performs.**



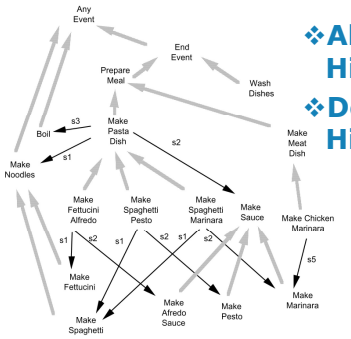
Plan Recognition for Interface Agents

- **We have to consider several issues:**
  - The uncertainty related to the moment in which the user starts a new plan
  - Overloaded tasks
  - Noisy tasks
  - Interruptions and plan abandonment
  - Multiple interleaved goals
  - Multiple plans
  - Adaptation to the user

Approaches to Plan Recognition

- **Consistency**
  - Determine which of an input set of goals is consistent with the observed tasks
  - A goal G is consistent with a task sequence T if T might have been executed in service of G
- **Probabilistic**
  - Select as a candidate intention that with the higher probability in light of the evidence at each moment

Plan library for a Consistency approach

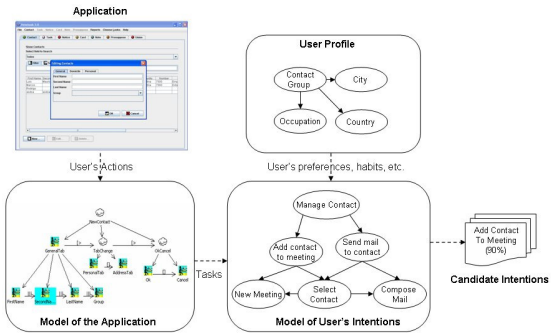


- **Abstraction Hierarchy**
- **Decomposition Hierarchy**

Probabilistic approach

- **Model for the detection of the user's intentions**
  - Used by an interface agent as the context in which the user is moving through
    - Assist the user in the context of his intention
    - Finding appropriate moments to initiate an interaction with the user
  - Three basic steps
    - Observe the user's interaction with the application
    - Known the task the user is performing in the application
    - Infer the user's intention
      - Adapted to each particular user

Probabilistic approach - Overview

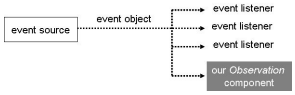


Monitoring user actions

- **We should not modify the application in order to detect user actions**
  - Dependent on the programming language
  - JAVA
  - Example

Monitoring user actions

- **Any number of objects can register as listeners on an event source**



Model of the application

- **Task Model**
  - Representation used to specify the tasks that the user can perform in the application.
  - Hierarchical structure
  - Temporal restrictions between tasks
  - ConcurTaskTrees
    - Rich set of operators
    - Task model simulator

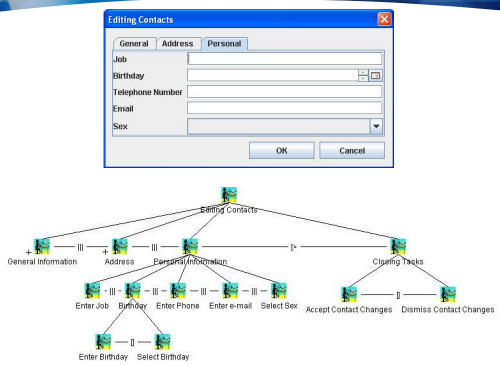
ConcurTaskTrees

- **Binary operators**
  - Choice: T1 [] T2
  - Order Independence: T1 |=| T2
  - Concurrency: T1 ||| T2
  - Concurrency with information exchange: T1 |[]| T2
  - Deactivation: T1 [> T2
  - Suspend / Resume: T1 |> T2
  - Enabling: T1 >> T2
  - Enabling with information exchange: T1 []>> T2
- **Unary operators**
  - Optional: [T1]
  - Iterative: T1*

ConcurTaskTrees

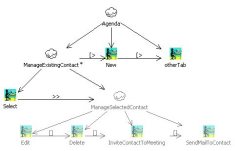
- **Categories of tasks**
  - Define the allocation of their performance
    - User Task
    - Application Task
    - Interaction Task
    - Abstract Task
- **Types of tasks**
  - Groups tasks according to their semantics
  - Each category has it owns types
    - For example, for interaction tasks: Selection, Control and editing

Model of the application



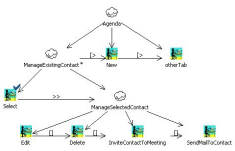
Model of the application

- **Operators allow the agent to know which tasks are *expected***
- **Task model simulator**
  - After each action performed by the user, the agent set the corresponding task as *performed* in the task model
  - According to the operators, the expected task list is updated



Model of the application

- **Operators allow the agent to know which tasks are *expected***
- **Task model simulator**
  - After each action performed by the user, the agent set the corresponding task as *performed* in the task model
  - According to the operators, the expected task list is updated



Model of user intentions

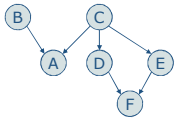
- **An interface agent should be able to detect higher level goals of the user**
  - Enhance and personalize the interaction with the user
  - Adapt its behavior to the user's needs
- **Model of user intentions**
  - General information about the intentions users can pursue in the application
  - Specific information about the habits and preferences of a particular user

Model of user intentions

- **Intention Graph**
  - models the intentions the user can pursue in the domain
  - represents the influence that a set of tasks has in the confidence the agent will have in any intention that the user could be pursuing.
  - materialized by a Bayesian Network
    - Knowledge representation capable of capturing and modeling dynamically the uncertainty of user-application interactions

Bayesian networks

- **Probabilistic knowledge representation used to model uncertain information**
  - DAG structure
    - Nodes → random variables
      - Mutually exclusive set of states
    - Arcs → causal relations
  - Strength of relations are encoded in conditional probability tables (CPT)



Bayes Theorem

- **Bayes' rule tells us how to update our belief about a hypothesis Vi in the light of new evidence Vj.**

$$p(Vi/Vj) = \frac{p(Vj/Vi)\,p(Vi)}{p(Vj)}$$

- **p(Vi|Vj) is calculated by multiplying our prior belief p(Vi) by the likelihood p(Vj|Vi) that Vj will occur if Vi is true**
- ***Evidence*: collection of findings on some variables**

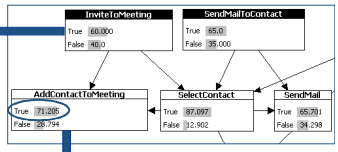
Intention Graph

- **Intention Graph G=<V,A,P,F,T>**
  - A set of variables V, where each variable can be of the type:
    - Task: a variable representing a task of the application
    - Goal: a variable representing a goal the user can pursue while using the application
    - Context: a variable representing attributes or properties of tasks in the application
  - A set of directed edges A between variables
  - Each variable has a finite set of mutually exclusive states
  - The variables together with the directed edges form a directed acyclic graph (DAG).
  - To each variable v in V with parents $v_1,\ldots,v_n$, there is attached the potential table P_v encoding $p(v|v_1,\ldots,v_n)$
  - A fading function F for evidence introduced in the network
  - A set of T traceable nodes, which is a subset of the nodes in V

Intention Graph

P(InviteToMeeting)=0.6

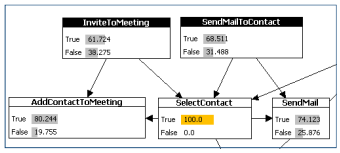
Confidence Level



P(AddContactToMeeting/SelectContact, InviteToMeeting) = 0.99
P(AddContactToMeeting/SelectContact, ¬InviteToMeeting) = 0.2
P(AddContactToMeeting/¬SelectContact, InviteToMeeting) = 0.5
P(AddContactToMeeting/¬SelectContact, ¬InviteToMeeting) = 0.01

Overloaded Tasks and Multiple Interleaved Goals

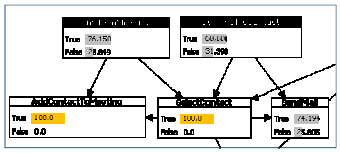
- **Overloaded task: a task that the user can perform to achieve multiple goals**



Agentes Inteligentes Plan Recognition

Overloaded Tasks and Multiple Interleaved Goals

- **Overloaded task: a task that the user can perform to achieve multiple goals**

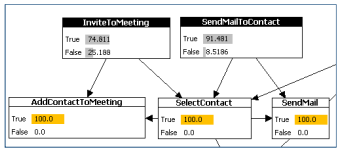


ISISTAN - UNICEN

Agentes Inteligentes Plan Recognition

Overloaded Tasks and Multiple Interleaved Goals

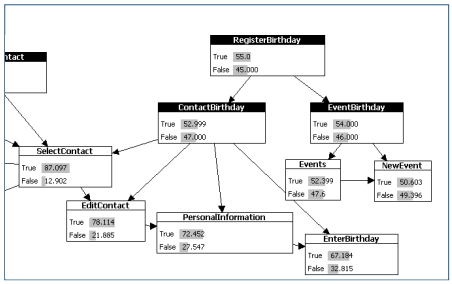
- **Overloaded task: a task that the user can perform to achieve multiple goals**



ISISTAN - UNICEN

Agentes Inteligentes Plan Recognition

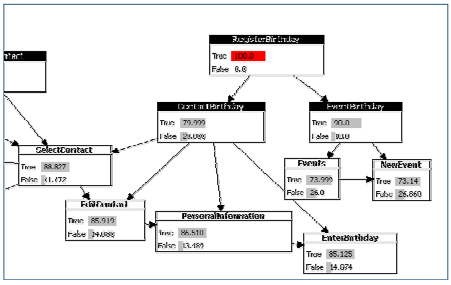
Multiple plans



ISISTAN - UNICEN

Agentes Inteligentes Plan Recognition

Multiple plans



ISISTAN - UNICEN

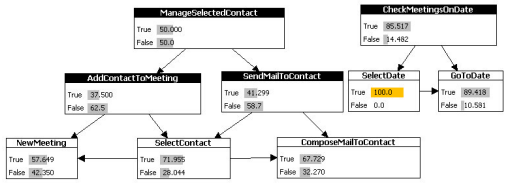
Agentes Inteligentes Plan Recognition

Noisy Tasks and changes in the user's goal

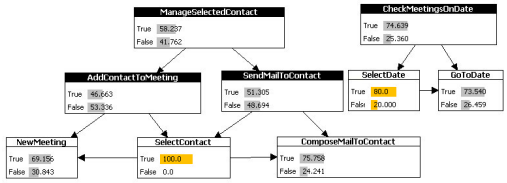
- **Fading function**
  - Gradually forget past observations
  - Ex.: decrement evidence by a constant factor
- **Soft evidence**
  - Use a probabilistic distribution to set evidence in Bayesian Networks

ISISTAN - UNICEN

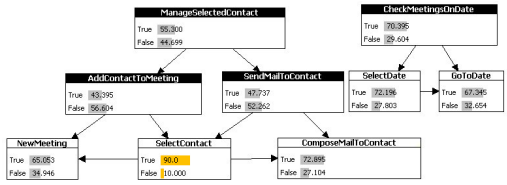
Noisy Tasks and changes in the user's goal



Noisy Tasks and changes in the user's goal



Noisy Tasks and changes in the user's goal



Personalization: Learning from the user

- **Adapt the process to a particular user of the application**
  - Adaptation of probabilities
  - Learning new relations

Adaptation of probabilities

- **Adapt the CPTs established by the designer to a particular user**
- **User Feedback declaring his intention**
- **Fractional updating** [Jensen2001]
  - Statistical approach
  - Experience count

$$p(N = k \mid conf) = \frac{n_k + y_k.z}{count_{old} + z}$$

Adaptation of probabilities

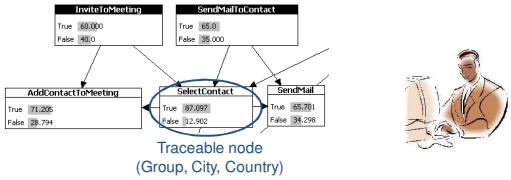
- **N=node**
- **k= state**
- **conf = pa(N) configuration**
- **z = probability of conf**
- **n_k = count of N being in state k**
- **y_k = probability of query N being in state k**
- **$count_{old}$ = previous count of experiences for n being in state k**

$$p(N = k \mid conf) = \frac{n_k + y_k.z}{count_{old} + z}$$

Learning new relations

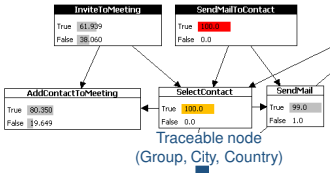
- **Use the attributes of the tasks performed by the user to build an interaction history**
- **Traceable node: a task node of the Intention Graph in which we want to register the values taken by such attributes**
- **Find new relations between these attributes and the nodes in the Intention Graph**
  - Batch learning and Parametric learning for Bayesian Networks.

Learning new relations



Traceable node
(Group, City, Country)

Learning new relations

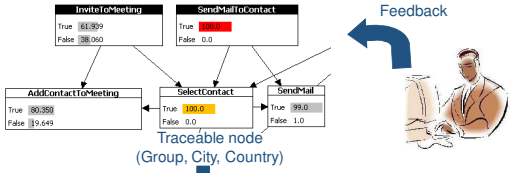


Feedback

Traceable node
(Group, City, Country)

```
Friends, Tandil, Argentina, False, True
Work, BsAs, Argentina, True, False
...
```


Learning new relations

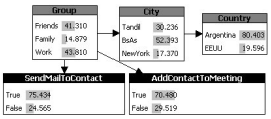


Feedback

Traceable node
(Group, City, Country)

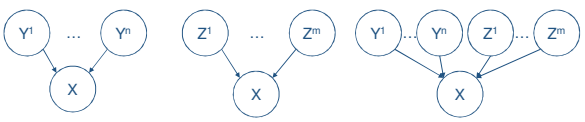
```
Friends, Tandil, Argentina, False, True
Work, BsAs, Argentina, True, False
...
```

Batch learning
+ Parametric learning



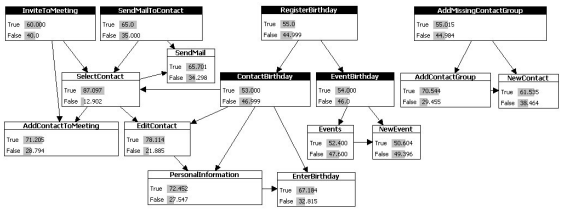
Learning new relations

- **Merging the learnt network into the Intention Graph**



$$p(X_j \mid Y_{1_{j1}},...,Y_{n_{jn}},Z_{1_{j1}},...,Z_{m_{jm}}) = p(X_j \mid Y_{1_{j1}},...,Y_{n_{jn}}).\prod + p(X_j \mid Z_{1_{j1}},...,Z_{m_{jm}}).(1-\prod)$$

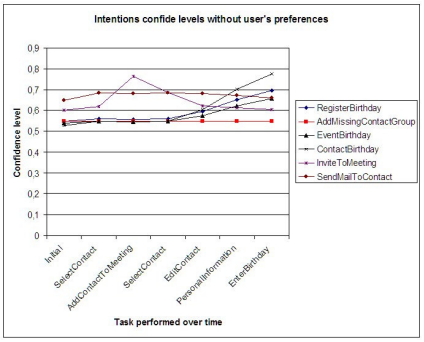
Example 1



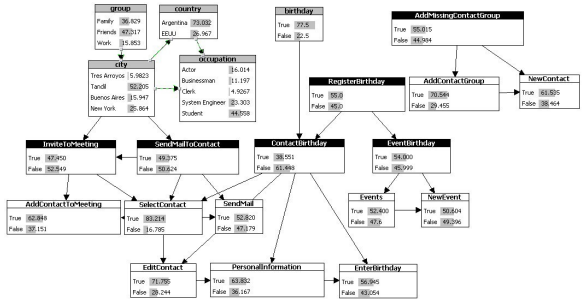
1. SelectContact
2. AddContactToMeeting
3. SelectContact
4. EditContact
5. PersonalInformation
6. EnterBirthday

1. Organize a meeting with John Smith
2. Register Mary's Birthday

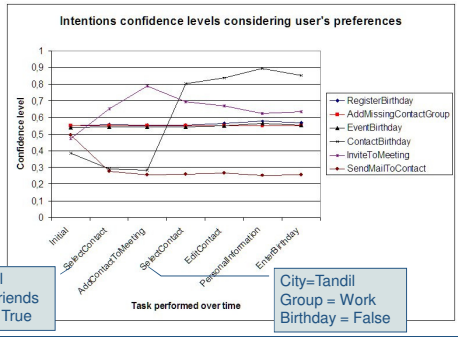
Example 1



Example 2



Example 2



City=Tandil
Group = Friends
Birthday = True

City=Tandil
Group = Work
Birthday = False

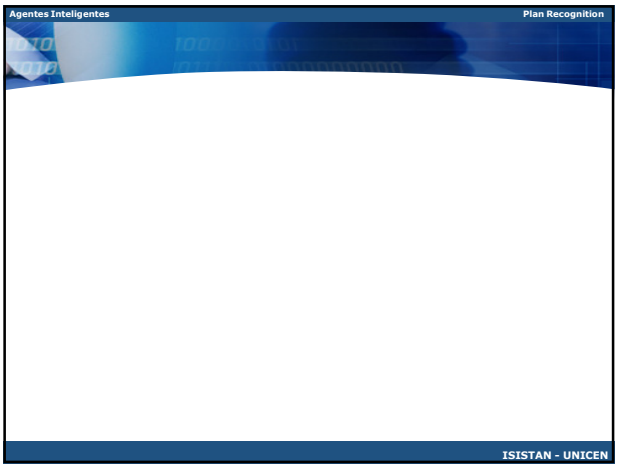
Summary

- **Inferring the goal of the user from observations of his actions**
  - A problem of inference under conditions of uncertainty.
- **Non-probabilistic approaches can not decide to what degree the evidence supports any particular goal hypothesis.**
  - important issue to consider so that the agent could be able to rank different possible explanations supported by the set of performed actions.
- **We presented a probabilistic approach able to**
  - deal with some common problems of plan recognition
  - adapt to a particular user of the software application

Questions

References

- [Bauer, 1996] Bauer, M. (1996). Acquisition of user preferences for plan recognition. In SpringerWien, editor, Proceedings of the Fifth Internataional Conference on User Modeling (UM '96), pages 105112, New York.
- [Brown, 1998] Brown, S. (1998). A Decision Theoretic Approach for Interface Agent Development. PhD thesis, Faculty of the Graduate School of Engineering of the Air Force Institute of Technology Air University.
- [Charniak and Goldman, 1993] Charniak, E. and Goldman, R. P. (1993). A bayesian model of plan recognition. Articial Intelligence, 64(1):5379.
- [Goldman et al., 1999] Goldman, R., Geib, C., and Miller, C. (1999). A new model of plan recognition. In Proceedings of the 15th Annual Conference on Uncertainty in Articial Intelligence (UAI-99), pages 24525, San Francisco, CA. Morgan Kaufmann.
- [Horvitz et al., 1998] Horvitz, E., J., B., Heckerman, D., Hovel, D., and Rommelse, K. (1998). The lumiere project: Bayesian user modeling for inferring the goals and needs of software users. In In Proceedings of the Fourteenth Conference on Uncertainty in Articial Intelligence, pages 256265, Madison, WI.
- [Huber et al., 1994] Huber, M. J., Durfee, E. H., and Wellman, M. P. (1994). The automated mapping of plans for plan recognition. In de Mantaras, R. L. and Poole, D., editors, Proceedings of the 10th Conference on Uncertainty in Arti- cial Intelligence, pages 344351, San Francisco, CA, USA. Morgan Kaufmann publishers Inc.: San Mateo, CA, USA.
- [Jensen, 2001] Jensen, F. V. (2001). Bayesian Networks and Decision Graphs. Springer Verlag, New York.
- [Kautz, 1987] Kautz, H. (1987). A Formal Theory of Plan Recognition. PhD thesis, Dept. of Computer Science, University of Rochester.
- [Kim and Valtorta, 2000] Kim, Y.-G. and Valtorta, M. (2000). Soft evidential up-date for communication in multiagent systems and the big clique algorithm. Tech-nical report, Department of Computer Science and Engineering, University of South Carolina.
- [Lesh, 1999] Lesh, N.; Rich, C. S. C. (1999). Using plan recognition in human-computer collaboration. In Seventh International Conference on User Modeling, Ban Canada.
- [Maes, 1994] Maes, P. (1994). Agents that reduce work and information overload. Communications of the ACM, 37(7):3040
- [Pearl, 1988] Pearl, J. (1988). Probabilistic Reasoning in Intelligent Systems. Morgan Kaufmann Publishers, San Mateo, California.


Agentes Inteligentes
Plan Recognition
ISISTAN - UNICEN