

Context-Based Method Using Bayesian Network in Multimodal Fission System

Atef Zaguia

*Department Of Computer Science, College of computers and information technology,
Taif University, Taif, Saudi Arabia
E-mail: zaguia.atef@tu.edu.sa*

*MMS Laboratory, Université du Québec, École de technologie supérieure
1100, rue Notre-Dame Ouest, Montréal, Québec, H3C 1K3 Canada
E-mail : atef.zaguia.1@ens.etsmtl.ca*

Chakib Tadj

*MMS Laboratory, Université du Québec, École de technologie supérieure
1100, rue Notre-Dame Ouest, Montréal, Québec, H3C 1K3 Canada
E-mail : ctadj@ele.etsmtl.ca*

Amar Ramdane-Cherif

*LISV Laboratory, Université de Versailles-Saint-Quentin-en-Yvelines, Paris, France.
E-mail : rca@prism.uvsq.fr*

Received 4 May 2015

Accepted 27 September 2015

Abstract

The current technological advancement has created the need to produce machines more powerful, easy to use and to meet the needs of users. To achieve this goal, multimodal systems have been developed to combine multiple modalities depending on the task, preferences and communicative intentions of the users.

In this paper, we present a new approach to overcome uncertain or ambiguous knowledge during the fission process in multimodal system. This approach is context-based method using Bayesian network. We also present a modular and distributed architecture, which is very useful in multimodal systems.

Keywords: Multimodal fission; Bayesian network; Modality; Pattern; Interaction context;

1. Introduction

The Human-machine Interaction is the discipline devoted to the design, implementation and evaluation of interactive systems for a human users as well as the study of major phenomena surrounding them. An interactive system is a system whose operation depends on information provided by an external environment that does not control this system. Interactive systems are also known as open systems - or autonomous - the functioning of which can be completely illustrated by algorithms. **Refs 1 and 2**

Moreover, Interactive systems are the outcome of scientific needs. Their popularity is due to their use by a great number of people. This fact led researchers to

develop systems / machines according to the user's need and spread their use on a high scale. The current technological advancement has created the need to produce more powerful machines, easy to use and to meet the needs of users. To achieve these objectives, these machines should interfere in a harmonious way with the user. This could be made possible only if the systems are able to understand human communication. Human communications is conducted via many naturel modalities such as speech, gesture, eyes gaze and facial expressions. Inspired by human communications, multimodal systems were developed to combine many modalities according to the task, preferences and communication intentions. End users could take advantage of natural modalities (e.g. eye gaze, speech,

gesture, etc.) to communicate or exchange information with applications/machines. These machines, integrated with natural modalities, is an effective solution for users who cannot use a keyboard or a mouse, on users who have visual handicap, on mobile users equipped with wireless telephone/mobile devices, on weakened users, etc.

Therefore the machines must be able to recognize the user's commands and communicate similarly to human with the use of different ways of communication such as words, gestures, voice, etc.

Multimodal systems are an effective solution to this problematic. They allow the use of different modalities (gesture, vocal, etc.) to interact with machine, system, applications, etc.

Nowadays, we are seeing a movement towards convergence of human-machine / machine-machine interactive systems to multimodal systems that enhance the interaction. These systems use the adequate modalities according to the user's preferences, his/her skill level and the nature of the task.

Generally, these systems have a process of understanding. They must also have the ability to interpret data from multiple input modalities generated from sensors and devices (camera, microphone, etc.). This is known as fusion process **Refs 3 and 5**. The resulting command is then executed in the output device (screen, speakers, projector, etc.). This is known as fission process, the role of the fission module is to subdivide the requests/commands made by the user to elementary subtasks, then associate them to the appropriate and available output modalities **Refs 5-7**.

A well-known example of these systems is the Bolt system "Put That There" **Ref.8**, where the author used gesture and speech to move objects. Another simple example of multimodal system is the use of the ringer and brightness to indicate a call or GPS which provides visual and audible indications.

These systems improve the recognition and the understanding of the environment command (user, robot, machine, etc.) by the machine.

This paper focuses on the fission process. We present a new methodological solution by modeling an architecture that facilitates the work of a fission module, by defining an ontology that contains different applicable scenarios and describes the environment, where a multimodal system exists. We detail in this article the use of a probability method, namely the

Bayesian network (BN), to perform multimodal fission. The BN is helpful in the case of ambiguity or uncertainty.

This paper discusses these solutions by highlighting the architectural design of the proposed solution. The rest of this paper is organized as follows. Section 2 presents the most important research works related to ours. Section 3 presents the proposed architecture. Section 4 describes the fission algorithm. Section 5 discusses the BN. Section 6 presents a concrete simulation example using CPN-Tools. The paper is concluded in section 7.

2. Related Work

The Multimodal systems represent a remarkable deviation from the use of conventional systems, such as windows and icons, to a man / machine interaction, providing to the user more naturalness, flexibility and portability.

The first multimodal system was created in 1980 by Richard Bolt "Put-That-There" **Ref.8**. The system is equipped with a microphone and a screen. It allows moving or changing the display of objects on the screen, using the voice accompanied by pointing on the screen. Since Bolt's work, the academic world has provided models and designs offering multimodal interaction techniques **Refs 9-12**.

The multimodal system is mainly composed of two indispensable modules: fission module **Ref.13** and fusion module **Ref.14**.

When two or more modalities are invoked at the same time (e.g. speech and clicking a mouse button), the user invokes complementarities of these modalities. This provides a rationale for the invocation of multimodal fusion and the fusion engine which is responsible for determining the meaning of such complementarities.

The fission is the way to segment the data that will be presented to the user according to the available modalities and context **Ref.15**. The fission module is a crucial component of multimodal systems. However, most research in multimodal systems focus more on the fusion than the fission. This point is supported by "There isn't much research done on fission of output modalities because most applications use few different output modalities therefore simple and direct output mechanisms are often used"⁵ and by "Multimodal fission is a research topic that is not often addressed in the scientific community."¹⁴

This module is the main subject in our work. We focus specially on 1) the services connected to the output: multimodal fission and, 2) the creation of a multimodal interaction system.

The few examples presented in the literature about the fission process are very simple or question-response based process, that provide limited choices to the user. The user is therefore restricted to some particular command (predefined fission). For example, in the system presented in **Ref.17**, the fission process is very simple. It detects the state of the driver. It captures information from sensors installed in the car and then it tests if the values entered are in a specific range, and through this test the system will generate alerts like sonar alerts or vibration of the wheel.

Most of the architectures presented for multimodal systems are dedicated to very specific modalities **Ref 18-20**. Also, most of the multimodal systems presented in the literature don't mention solutions to overcome the problem of uncertainty or ambiguity during the fission or fusion process. In **Ref.17**, they use the BN in the fusion process to model human fatigue and to predict fatigue based on the visual cues obtained when a driver is driving a car. The BN presented by the authors don't deal directly with fusion process but just to determine a parameter (level of fatigue) that affects the decision in the fusion process

In this work we introduce a probabilistic reasoning to integrate uncertain and ambiguous knowledge. We use BN that we adapted using the contextual information

such as temperature, history, user status, etc. to resolve the problem of ambiguity and uncertainty in the fission system.

3. Architectural design

In this section, we present our proposed architectural design as a solution to the described problems of previous architectures.

A general overview of the architectural design is shown in Fig.1. The proposed system architecture is modular and distributed. All the components can be installed in different places in the network (robot, computer, server, etc.). The architecture is composed mainly of 7 modules/components. These components are as follows:

Command Extraction: this module has in input different XML files from applications. The output corresponds to the command that our system will process. The system interacts with many applications. It exchanges data (information about environment, command, user preference, etc.) using XML files. The purpose of this module is to extract the command from the XML files and send it to the fission module;

Context Acquisition Module: the goal of this module is to select the adequate modality (ies) according to the user's preference, status of media and the status of the environment.

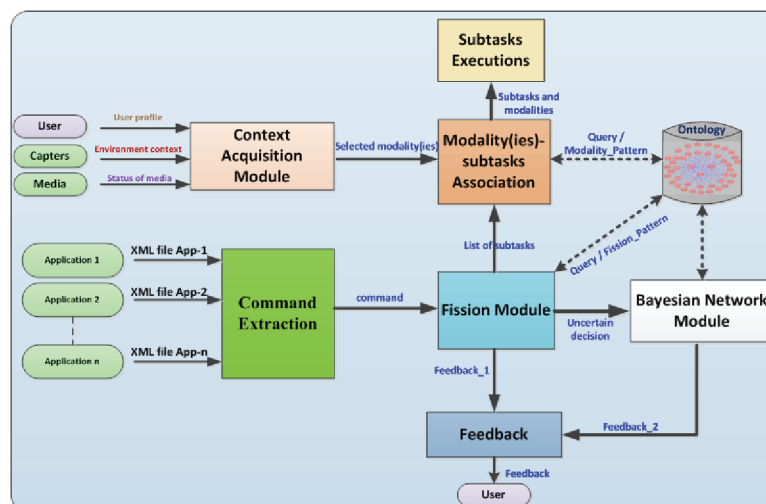


Fig.1. General approach of multimodal fission system

This module has in input:

- The state of the environment obtained from the captors. It selects the appropriate modalities depending on information collected from the captors such as the noise level. It is understood that the use of the audio modality is affected by this information. For example, if the noise level is high, the use of the audio modality is disabled. This is called environment context **Ref.21** ;
- The user location and the status of the user. It defines the user's ability to use certain modalities. For example if the user is blind the display modality is disabled and if the user is in a library the audio modality is disabled. This is called user context **Ref.21** ;
- The state of the media. This allows to determine the capacity and the type of system that we use. For instance if the battery of the cell phone is low then the system disables the audio modality and decreases the level of brightness. This is called system context **Ref.21**.

Modality(ies)-subtasks Association: the goal of this module is to select the adequate modality (ies) for a given subtask. Their inputs are the selected modality (ies) and the list of subtasks obtained from Context Acquisition Module and the Fission Module, respectively. The output contains the list of subtasks with the adequate modality (ies). This module sends queries to the ontology to find the matching pattern **Ref.22** of subtask selection. In our case, pattern is defined by two parts: problem that presents the model of the command and solution that contains the suitable subtasks;

Ontology: is the knowledge base that describes every detail in the environment (objects and events, etc.). It also contains all possible patterns **Ref.6** of the subtask selection and the modalities selection;

Fission module: is the crucial module in our architecture. It decomposes a command to elementary subtasks. This module has in input the command obtained from Command Extraction. The output contains:

- the list of subtasks which are then sent to the subtasks executions module,
- or a feedback message if the command is incorrect,
- or the command to the BN module in the uncertain case.

This module is detailed in section 4.

For more details about the previous modules, the reader can refer to **Ref.6** and **Ref.3**.

Bayesian Network Module: in case of ambiguity or uncertainty, this module allows interaction with the ontology to make a decision using a probabilistic method. The input of this module is the ambiguous command. The output provides the decision to the feedback module.

4. Fission algorithm

The fission process is the crucial part of the multimodal system. The fission is known as the way to subdivide commands to elementary subtasks and present them to the user according to the available modalities and context. Consequently, we can say that the goal of multimodal fission is to pass from a presentation independent of modalities to a coordinated and coherent multimodal presentation.

In general, the fission rule is simple (Fig. 2): if a complex command (CC) is presented, then a list of subtasks with suitable modalities (and its parameters) is deduced.

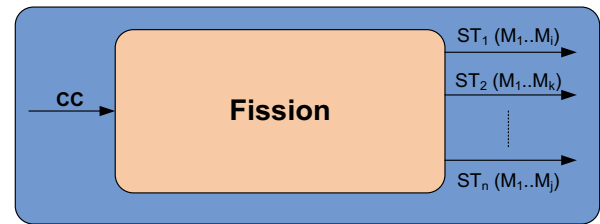


Fig. 2. Fission process

Multimodal fission can be represented by the function:

$$f: F \rightarrow ExK$$

$$\forall cc \in F, \exists ST_i \in K \text{ and } MO_j \in E,$$

$$f(ST_i, MO_j) = cc$$

$$\text{with } i \in [1..n] \text{ and } j \in [1..m]$$

$$f: CC = \sum_{i=1}^n ST_i(\{\cup_{j=1}^k MO_j\}, \{\cap_{j=1}^l MO_j\}) \quad (1)$$

With:

ST_i = sub-task i .

MO_j = output modality j .

cc = complex command.

l and k are different from m and n because it depends on the sub-tasks. For example, for some sub-tasks we will use only two terms even if we have three available modalities.

In equation (1), symbol \cup indicates that we can use either one or several modalities to present a sub-task. For example, if we present a text to the user, we use audio or display. The symbol \cap indicates that we use the available modalities together to present a sub-task. Stages of fission process are described in Fig. 3. The fission process starts when the system receives an XML file from a given application.

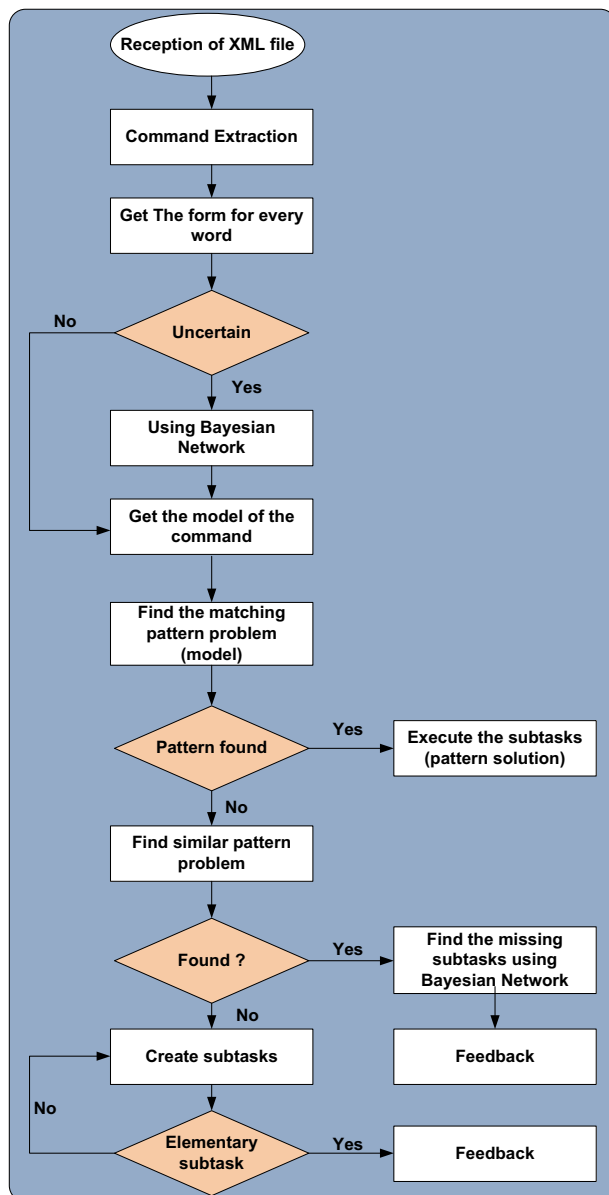


Fig. 3. Fission algorithm

Then it extracts the command from the XML file and the system interacts with the ontology to get the meaning for every word (this part is detailed in Ref.3).

In the case of ambiguity or uncertainty decision the system uses BN to resolve the problem (this part is detailed in section 5) or else it gets the model of the command from the ontology. Then the system sends a query (containing the problem parameters) to the ontology to find the matching pattern. Pattern is defined with two parts: problem and solution. In our case the pattern problem presents the model of command and the pattern solution the list of subtasks. If the pattern is found the system gets the list of subtasks and selects for every subtask the adequate modality (ies) (detailed in Ref.6) or else the system will try to find a similar command in our ontology. If not found, the system creates elementary subtasks and asks a feedback from the user, or else the system finds the missing subtask using a BN (detailed in section 3) and asks a feedback from the user.

5. Bayesian network module

BN are particularly suitable for “collecting and representing knowledge on uncertain domains but also enable to perform probabilistic calculus and statistical analyses in an efficient manner. The difference of BN, in comparison with other classical methods, is their polyvalence. They allow dealing with issues such a prediction or diagnosis, optimization, data analysis of feedback experience, deviation detection and model updating “**Refs 23 and 24**

Bayesian model has several important advantages that make it suitable for our purpose. We employ the Bayesian networks to determine, in case of uncertainty, the suitable subtasks to select during the process of the multimodal fission. Therefore the fission process will be autonomous to take decision in case of uncertainty (this case will be illustrated with reel example in section 6.1) and maintain the knowledge base updated in case of missing subtasks (self-learning). (This case will be illustrated with reel example in section 6.2) **Ref.25**

As we mentioned in the previous section, during the fission process, the system should make a decision between several equal choices. In uncertain or ambiguous cases, the BN is an effective solution to solve this problem.

In this section, we present the definition of the BN, then we present an example of how it is used in data mining and then how we adapt this method to use it within the context to overcome the problem of uncertainty in multimodal fission.

5.1. Definition

A BN provides a mechanism for graphical representation of uncertain knowledge and for inferring high-level activities from the observed data.

A BN is a graph in which nodes represent random variables and the links are influences between variables. The graph is acyclic: it contains no loop. **Ref.26.**

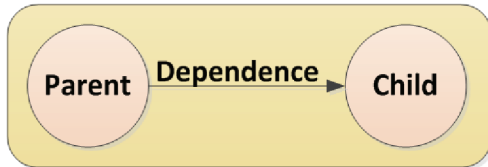


Fig. 4. Relationship between two nodes in a BN

The arc produces a probability distribution where the parent has the a priori probability $P(p)$, and the child conditional probability $P(c|p)$ (Fig. 4.).

BN are used in many fields for diagnosis (medical and industrial), risk analysis, spam detection and data mining.

5.2. BN in Data Mining

For an effective research, we use a semantic Bayesian graph that combines a semantic inference and probability.

As we can see in Fig. 5, a semantic Bayesian graph consists of several layers of nodes: queries, keywords, concepts and target resources.

At the first level of abstraction, keywords are used in user queries. Each keyword is connected to one or more concepts with a certain probability. Finally, each concept refers to a resource. The relationship between concepts and resources are semantic links.

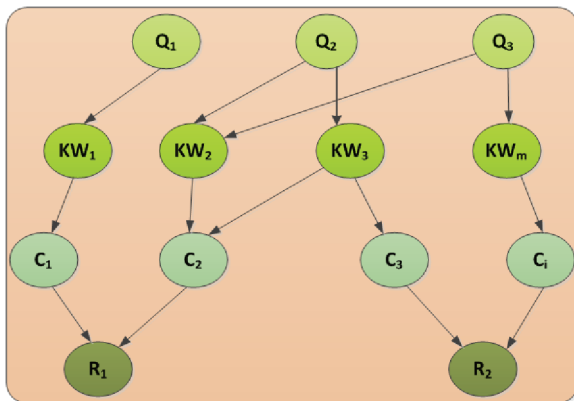


Fig. 5. Semantic Bayesian network

The search of resources based on a semantic Bayesian graph is performed as follows:

First, the query is parsed and keywords are selected. Then the probability layer between keywords and concepts layer is calculated:

$$P(C|M) = \frac{P(M|C)P(C)}{P(M)} \begin{cases} C: \text{concepts} \\ M: \text{keys words} \end{cases} \quad (2)$$

$$M = m_1, m_2, \dots, m_n \quad n: \text{number of keys words}$$

Where:

- $P(C|M)$: a posteriori probability ;
- $P(M|C)$: likelihood ;
- $P(M)$: evidence ;
- $P(C)$: a priori probability.

The idea in data mining is as follows: for each concept, its probability is calculated based on the conditional probabilities of the keywords of the user query $P(M|c)$. Thereafter, it chooses the concept that has the highest probability. Since each concept's node points to a target resource, then the resource related to the chosen concept is returned as a response.

The conditional probability $P(M|c)$ is assigned manually by consulting an expert in each field and $P(c)$ generally equals 1 to give the same weight to each concept.

From this perspective, we present in the next paragraph the adaptation of this method to our case with the use of context.

6. BN applied in fission process

We can use the same concept to select data from the ontology in the case of uncertainly or ambiguity. We can represent the adaptation of the BN with context by the equation (4.3)

$$Con_i \rightarrow \prod_{j=1}^m (C_j, P_j) \quad (3)$$

With $i \in [1..n]$

n presents the number of context.

m presents the number of ambiguous concepts.

Con = context, C = concept and P = probability.

Each context is connected to one or more concepts with a certain probability.

Here we explain our method using two different applications: GPS (by 1 example) and robot control (by 2 examples).

6.1. GPS example

Suppose the user asks the GPS: “I want to go to Montreal”. As we can see in the ontology, the user may mean one of three possible concepts: C1 {City}, C2 {Restaurant} and C3 {Street} (Figure 6).

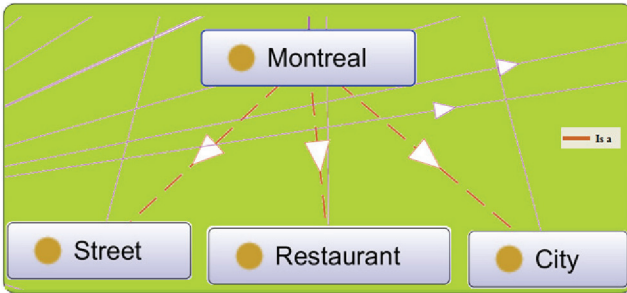


Fig. 6. Montreal concept

In data mining, they use keywords of the command to select the appropriate resources. However, in this case the keywords {go, Montreal} are useless because all of them refer to destination. We can overcome this problem by taking into account the context instead of keywords such as in data mining. In this work we define context as: user context: user status, location, time and history. Context information is collected from sensors or captors available in the user's environment. Suppose that the captured context is:

- the actual location is Paris ;
- the user is not hungry ;
- the actual time is 13:00.

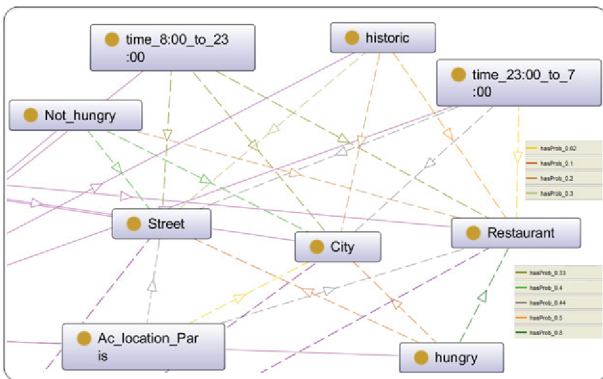


Fig. 7. Bayesian Network for Montreal

Assume that the likelihood probabilities, usually estimated by an expert, are as follow (Fig.7):

$$P(\text{not Hungry}|C1) = 0.4$$

$$P(\text{not Hungry}|C2) = 0.2$$

$$P(\text{not Hungry}|C3) = 0.4$$

$$P(\text{Ac_location_Paris} |C1) = 0.02$$

$$P(\text{Ac_location_Paris}|C2) = 0.44$$

$$P(\text{Ac_location_Paris}|C3) = 0.44$$

$$P(\text{historic}|C1) = 0.2$$

$$P(\text{historic}|C2) = 0.5$$

$$P(\text{historic}|C3) = 0.3$$

$$P(\text{time_8:00_to_23:00} |C1) = 0.33$$

$$P(\text{time_8:00_to_23:00} |C2) = 0.33$$

$$P(\text{time_8:00_to_23:00} |C3) = 0.33$$

Here we integrated many dynamic variables such as time, history, actual location, status of user, which can affect the decision. We can add more variables if we judge that it may to affect the result.

Then we calculate the a posteriori probabilities for every concept using equation (2).

$$P(C1|\text{Ac_location_Paris}, \text{historic}, \text{time_8:00_to_23:00}, \text{not hungry}) =$$

$$P(C1) \times \frac{P(\text{Ac_location_Paris}|C1) \times P(\text{historic}|C1)}{P(\text{Co})} \times$$

$$P(\text{not hungry}|C1) \times P(\text{time_8:00_to_23:00}|C1) =$$

$$1 \times \frac{0.4 \times 0.02 \times 0.2 \times 0.33}{P(\text{Co})} = \frac{5 \times 10^{-4}}{P(\text{Co})}$$

$$P(C2|\text{Ac_location_Paris}, \text{historic}, \text{time_8:00_to_23:00}, \text{not hungry}) =$$

$$P(C2) \times \frac{P(\text{Ac_location_Paris}|C2) \times P(\text{historic}|C2)}{P(\text{Co})} \times$$

$$P(\text{not hungry}|C2) \times P(\text{time_8:00_to_23:00}|C2) =$$

$$1 \times \frac{0.2 \times 0.44 \times 0.5 \times 0.33}{P(\text{Co})} = \frac{0.014}{P(\text{Co})}$$

$$P(C3|\text{Ac_location_Paris}, \text{historic}, \text{time_8:00_to_23:00}, \text{not hungry}) =$$

$$P(C3) \times \frac{P(\text{Ac_location_Paris}|C3) \times P(\text{historic}|C3)}{P(\text{Co})} \times$$

$$P(\text{not hungry}|C3) \times P(\text{time_8:00_to_23:00}|C3) =$$

$$1 \times \frac{0.4 \times 0.44 \times 0.3 \times 0.33}{P(\text{Co})} = \frac{0.017}{P(\text{Co})}$$

As mentioned previously, we set $P(C1) = P(C2) = P(C3) = 1$. We do so to give the same weight to every concept.

As we can see, $P(C3|Co)$ has the highest value. This means that the concept $C3 = \{\text{Street}\}$ is the most probable request the user meant. The system will seek feedback from the user “what is the number of the building in the Montreal Street?”

6.2. Robot control example 1

Suppose that the user gives an order to the robot “give me water”. As we can see in the ontology (Fig.8), we have three possible cases: add water in glass or add water in casserole or add water in pail (in real life we can have many other choices, but we limit it to what we have in our ontology). We are in the case of uncertainty. We use the BN to resolve this problem. As shown in

Figure 4.9, different contexts can influence the decision making. Each context is associated to all ambiguous concepts with a corresponding probability.

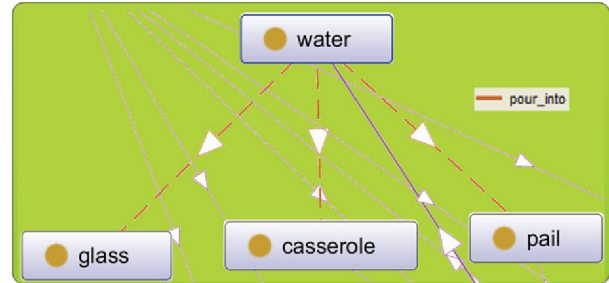


Fig. 8. Example water ontology

Using equation (1), we calculate the a posteriori probabilities for each concept {glass, casserole, pail}. Table 4.1 shows all likelihood probabilities.

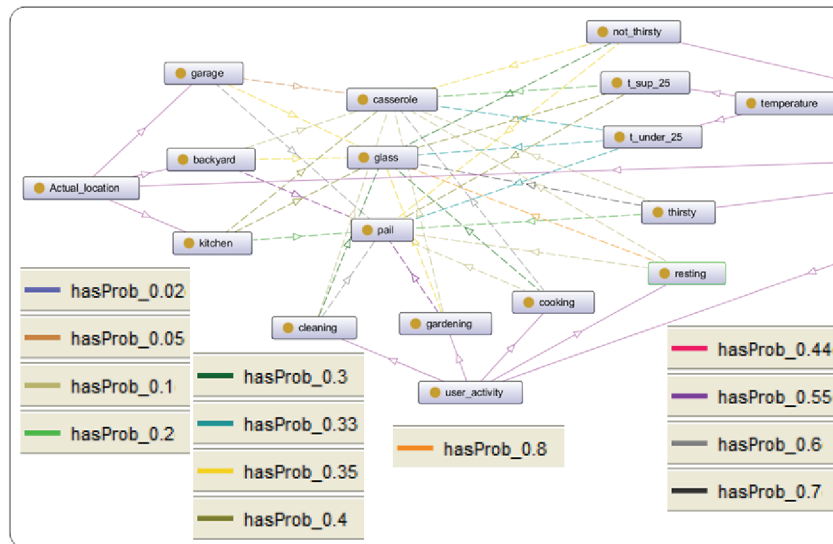


Fig. 9. Likelihood probabilities

Table 1. Probabilities of likelihood of concepts {pail, casserole, glass}

	C1 {pail}	C2{casserole}	C3 {glass}
garage	0.6	0.05	0.35
kitchen	0.2	0.4	0.4
backyard	0.55	0.1	0.35
thirsty	0.2	0.1	0.7
not thirsty	0.35	0.35	0.3
t_under_25	0.33	0.33	0.33
t_sup_25	0.4	0.2	0.4
cleaning	0.6	0.1	0.3
gardening	0.55	0.1	0.35
cooking	0.1	0.6	0.3
resting	0.1	0.1	0.8

Suppose that captors, installed in the robots detect that the user is in the backyard, he is not thirsty, the temperature is higher than 25 degrees and he is gardening.

We calculate the a posteriori probabilities for every concept using equation (4.1) as in the previous example.

$$P(C1|backyard, Not\ thirsty, t_sup_25, gardening) = 1 \times \frac{0.55 \times 0.35 \times 0.4 \times 0.55}{P(Co)} = \frac{0.042}{P(Co)}$$

$$P(C2|backyard, Not thirsty, t_sup_25, gardening) = 1x \frac{0.1x0.35 * 0.2 * 0.1}{P(Co)} = \frac{0.0007}{P(Co)}$$

$$P(C3|backyard, Not thirsty, t_sup_25, gardening) = 1x \frac{0.35x0.33 * 0.4 * 0.35}{P(Co)} = \frac{0.016}{P(Co)}$$

As we can see, $P(C1|backyard, Not thirsty, t_sup_25, gardening)$ has the highest probability. The system will choose the pail to bring the water.

In this stage, we presented two different ways to use the BN for two different applications. In the next example, we present the case of similar command and we show how the system finds the missing subtask as presented in section 4.

6.3. Robot control example 2

Assume that the user asks a robot “prepare a coffee with milk”. Suppose that we already have in our ontology a pattern for “prepare a coffee” that contains all necessary subtasks.

Here we have similar commands (in red) and we have to find the missing subtasks (“prepare a coffee with milk”).

In this case we select the adequate pattern for the command “prepare a coffee” and we find the missing subtask.

Milk presents a liquid in our ontology and we have already defined some possible elementary subtasks for liquid and objects as shown in Figure 10.

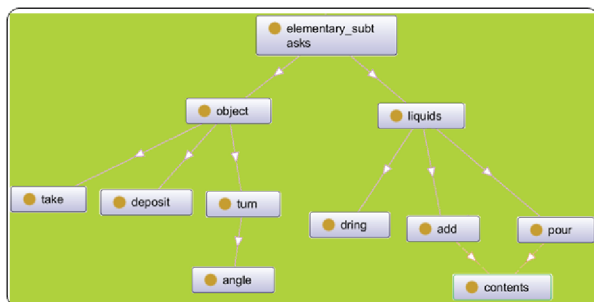


Fig. 10. Some possible elementary subtasks for liquid and objects

We have three possible decisions: either drink liquid or add liquid in content or pour liquid in content. To make a decision, the system uses BN (Fig. 11). We have three concepts: C1 {drink}, C2 {add} and C3 {pour}. We

have two keywords related to these concepts with and liquid. In this case we use the keywords as used in data mining. As shown in Fig. 11, the two keywords are related to the concepts with probabilities of likelihood:

$$\begin{aligned} P(liquid|C1) &= 0.5 \\ P(liquid|C2) &= 0.5 \\ P(liquid|C3) &= 0.5 \end{aligned}$$

$$\begin{aligned} P(with|C1) &= 0.15 \\ P(with|C2) &= 0.7 \\ P(with|C3) &= 0.15 \end{aligned}$$

$$\begin{aligned} P(C1|liquid, with) &= P(C1) \times P(liquid|C1) \times P(with|C1) \\ &= 1 \times 0.5 \times 0.15 = 0.075 \end{aligned}$$

$$\begin{aligned} P(C2|liquid, with) &= P(C2) \times P(liquid|C2) \times P(with|C2) \\ &= 1 \times 0.5 \times 0.7 = 0.35 \end{aligned}$$

$$\begin{aligned} P(C3|liquid, with) &= P(C3) \times P(liquid|C3) \times P(with|C3) \\ &= 1 \times 0.5 \times 0.15 = 0.12 \end{aligned}$$

Note that:

$$P(C1|liquid, with) < P(C3|liquid, with) < P(C2|liquid, with)$$

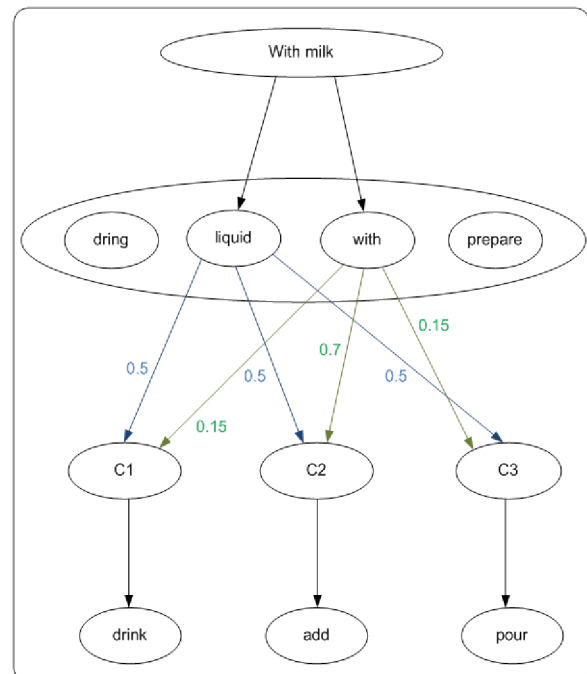


Fig. 11. Bayesian network presenting

The concept *add* is therefore chosen as it has the lowest probability. The system adds the subtask “add milk” to

the subtasks of the command “prepare a coffee” already existing in the ontology. The ontology is then updated with the new pattern.

Simulation

To understand the mechanism of fission using BN we demonstrate a simulation of a scenario.

In this scenario, we assume that in our multimodal system, a GPS indicates the directions to go to a specific location. For instance “I want to go to Washington”.

We first model and simulate the architecture defined in Fig. 1. The steps of each strategy are modeled using the colored Petri Net **Ref.27** formalism and simulated using the CPN Tools **Ref.28**.

Fig. 12 shows the general view of the architecture. It is composed mainly by 8 modules:

- **Generator**: this module generates events as random numbers to select a command in the *place* “command”.
- **T_parser**: this module decomposes the command into words.
- **T_Fission**: its role is to divide the command to elementary subtasks.
- **T_Grammar**: permits to verify the meaning and the grammar of the command.
- **Modality(ies) Selection**: this module allows to select the available modalities depending on the state of the environment.

- **T_Subtask-Modality_Association**: it associates for each subtask the appropriate modality (ies).
- **Ontology**: it is a container that stores the patterns as ontology concepts, the models and the vocabulary.

For more details about these modules, the reader can refer to **Ref.3**.

- **Bayesian Network**: this is the main module presented in this paper. Its role is to handle **uncertainty**.

As we can see in Fig. 13, the system receives the list of words of the command and the transition *list of the command*. The meaning for every word are retrieved from the ontology.

This is performed in the transitions *Decom List* and *meaning*. The role of *Decom List* is to send every word separately to the *meaning* transition. The *meaning* transition interacts with the ontology to get the meaning. As shown in Fig. 13, according to our example cited above, the place *meaning word*, *Washington* may have 7 possible solutions: {city, restaurant, basketball team, president, street, map}. In this case, the system won't be able to choose the right meaning.

Then in the transition *test of incertitude*, the system calculates the size of the meaning list. In our case, the size of *meaning* is greater than one. This is where the BN starts.

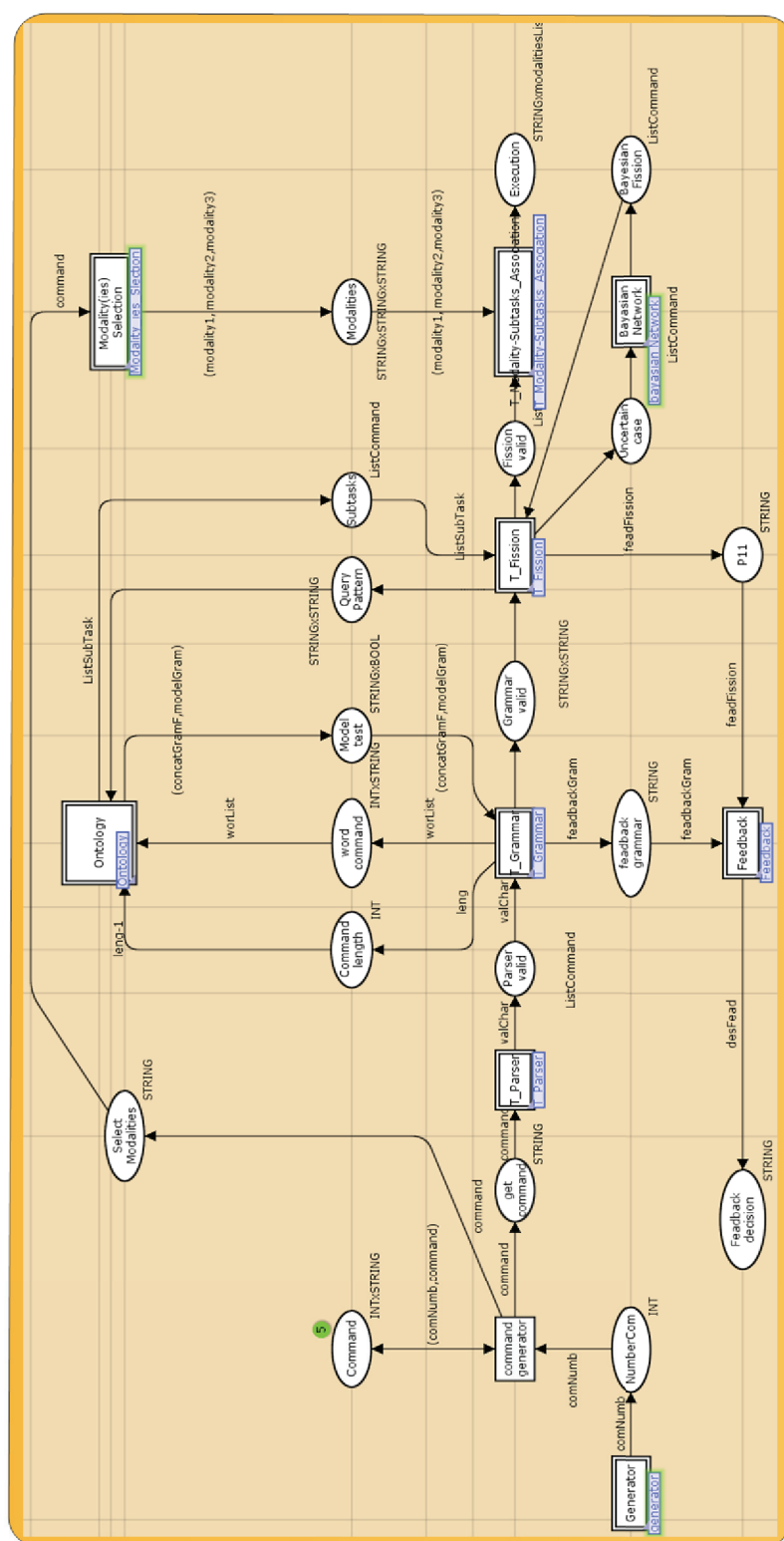


Fig. 12. General View of our architecture

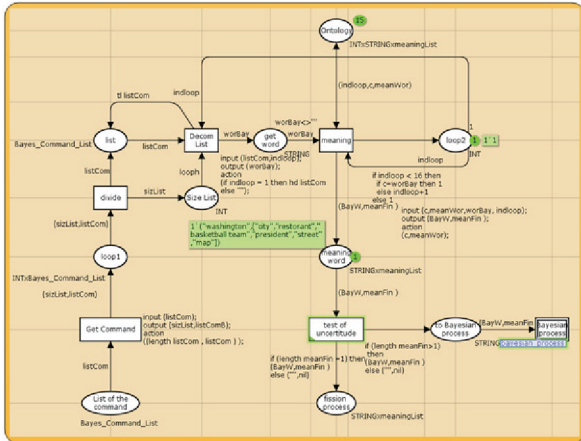


Fig. 13. Process of uncertainty detection

Fig. 15 shows the BN process. We have three main transitions: *Random context*, *probabilities of likelihood* and *calculate the a posteriori probabilities*.

The role of *Random context* is to simulate randomly information from captors. As shown in Fig. 14, the transition “collect information from captors” collects information from places: *location captors*, *user state*, *Application* and *time*. In our example, the captured information is:

Table 2. The captured information

Application	GPS
Time	17:00
actual location	Washington
user state	not hungry

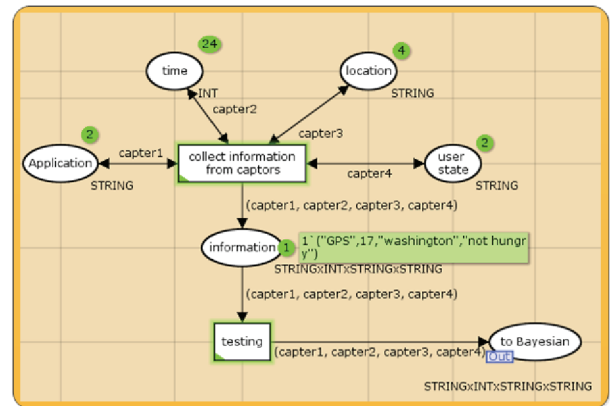


Fig. 14. Collect of information from captors

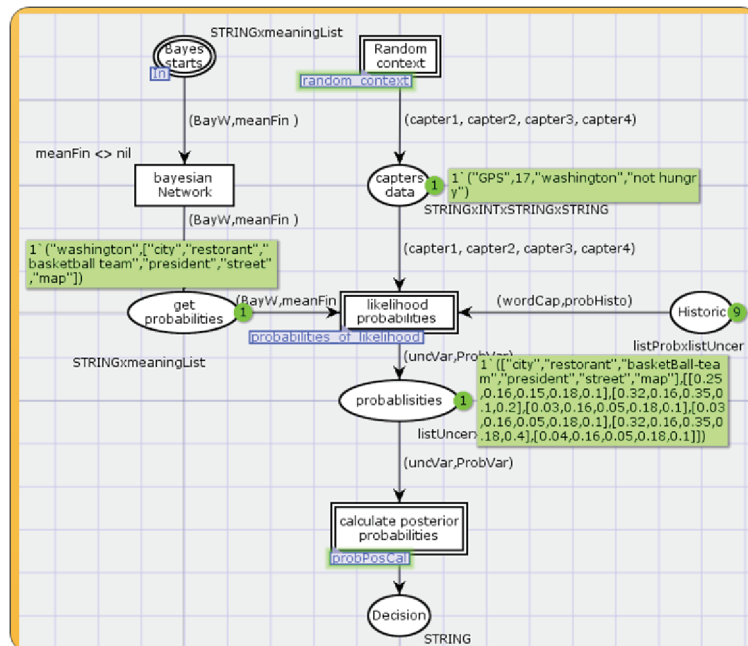


Fig. 15. Bayesian process

The transition *likelihood probabilities* contains different likelihood probabilities. As shown in Fig. 15, this transition has 2 inputs: from place *get probabilities* the word *Washington* and its meaning {city, restaurant, basketball-team, president, street, map} and from

captors data place the context {GPS, 17, Washington, not hungry} and for the output the probabilities of likelihood, in our case the probabilities are shown in Table 3.

Then the transition calculates the a posteriori probabilities and takes a decision about the most probable concept.

Table 3. Likelihood probabilities of concepts {city (Ci), restaurant (R), basketball-team (B-T), president (P), street (S), map (Ma)}

	GPS	17	Washington	NoHungry	history
Ci	0.25	0.16	0.15	0.18	0.1
R	0.32	0.16	0.35	0.1	0.2
BT	0.03	0.16	0.05	0.18	0.1
P	0.03	0.16	0.05	0.18	0.1
S	0.32	0.16	0.35	0.18	0.4
Ma	0.04	0.16	0.05	0.18	0.1

In Fig. 16, the system calculates the a posteriori probabilities for every concept using equation (1). The result obtained is (Fig. 16):

(["city","restorant","basketBall-team", "president", "street", "map"], [0.0001,0.0003,0.0,0.0,0.001,0.0]).

7. Conclusion

This paper presents an architecture for multimodal fission that makes use a Bayesian Networks to determine the list of subtasks to assign to different modalities.

The *fission* module allows segmenting the command generated by the machine to elementary subtasks and

presents them to the available output modalities. However, most of the multimodal systems studied in the literature use predefined modalities. The available architectures are targeted to specific applications. None of them seems to use any of the approaches belonging to probabilistic models when working with uncertainty, fuzzy or ambiguous real world data in multimodal systems.

In this paper we presented a new context-based method using Bayesian network to resolve the uncertainty problem during the fission process in a multimodal system. This method is useful, to overcome the problem of ambiguity or uncertainty. A concrete example was illustrated in order to show the effectiveness of the contribution, using CPN tool.

As a future work, we aim to develop a real application using methods presented in this paper.

Acknowledgements

We wish to acknowledge the funds provided by the Natural Sciences and Engineering Council of Canada (NSERC) which support the financial needs in undertaking this research work.

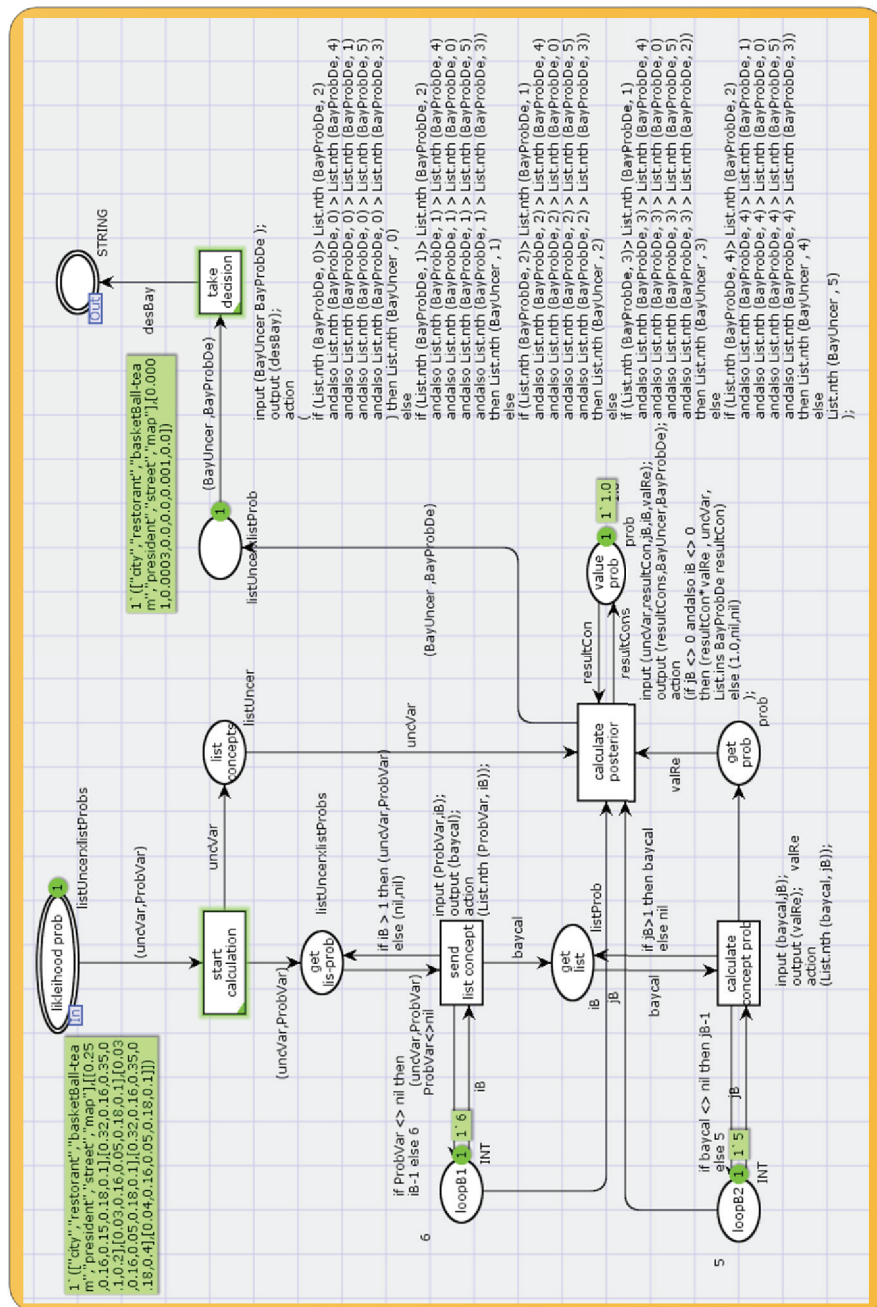


Fig. 16. Probabilities calculation

8. References

1. Thalmann, N. M., Thalmann, D., & Yumak, Z. (2014, November), Multimodal human-machine interaction including virtual humans or social robots, in *SIGGRAPH Asia 2014 Courses* (p. 14), ACM.
2. Jacko, J. A. (Ed.) (2012), *Human Computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emerging Applications*, CRC press.
3. Zaguia, Atef, Manolo Dulva Hina, Chakib Tadj et Amar Ramdane-Cherif, 2010b, "Using Multimodal Fusion in Computing and Information Sciences", vol. 1, no 2, p. 121-138.
4. Portillo, Pilar Manchón, Guillermo Pérez García et Gabriel Amores Carredano, 2006, "Multimodal fusion: a new hybrid strategy for dialogue systems", in *Proceedings of the 8th international conference on Multimodal interfaces (Banff, Alberta, Canada)*, p. 357-363.
5. Zaguia, Atef, Ahmad Wahbi, Chakib Tadj et Amar Ramdane-Cherif, 2013b, "Multimodal Fission For

- Interaction Architecture”, *Journal of Emerging Trends in Computing and Information Sciences*, vol. 4, no 1.
6. Zaguia, Atef, Ahmad Wahbi, Moeiz Miraoui, Chakib Tadj et Amar Ramdane-Cherif, 2013a, “Modeling Rules Fission and Modality Selection Using Ontology”, *Journal of Software Engineering and Applications*, vol. 7, no 6, p. 354-371.
 7. Costa, David, and Carlos Duarte, 2011, “Adapting Multimodal Fission to User’s Abilities”, in *Universal Access in Human-Computer Interaction, Design for All and eInclusion*, sous la dir. de Stephanidis, Constantine. Vol. 6765, p. 347-356. Coll. “Lecture Notes in Computer Science”: Springer Berlin Heidelberg. < http://dx.doi.org/10.1007/978-3-642-21672-5_38 >.
 8. Bolt, R., 1980, “Put-that-there”, Voice and gesture at the graphics interface, *ACM SIGGRAPH Computer Graphics*, vol. 14, no 3, p. 262-270.
 9. Jacob, Mithun George, Yu-Ting Li et Juan P Wachs, 2012, “Gestonurse: a multimodal robotic scrub nurse”, in *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, p. 153-154. ACM.
 10. Nordahl, Rolf, Stefania Serafin, Luca Turchet et Niels Christian Nilsson, 2012, “A multimodal architecture for simulating natural interactive walking in virtual environments”, *PsychNology*, vol. 9, no 3, p. 245-268.
 11. Oviatt, S., P. Cohen, Lizhong Wu, J. Vergo, L. Duncan, B. Suhm, J. Bers, T. Holzman, T. Winograd, J. Landay, J. Larson and D. Ferro, 2000, “Designing the user interface for multimodal speech and pen-based gesture applications: state-of-the-art systems and future research directions”, *Human-Computer Interaction*, vol. 15, no 4, p. 263-322.
 12. Meng, H., S. Oviatt, G. Potamianos and G. Rigoll, 2009, “Introduction to the Special Issue on Multimodal Processing in Speech-Based Interactions”, *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no 3, p. 409-410.
 13. Poller, Peter, et Valentin Tschernomas, 2006, “Multimodal Fission and Media Design”, in *SmartKom: Foundations of Multimodal Dialogue Systems*, edited by Wahlster, Wolfgang. Springer Berlin Heidelberg.
 14. Atrey, Pradeep K, M Anwar Hossain, Abdulmotaleb El Saddik and Mohan S Kankanhalli, 2010, “Multimodal fusion for multimedia analysis: a survey”, *Multimedia Systems*, vol. 16, no 6, p. 345-379.
 15. Nguyen, Laurent, Jean-Marc Odobez and Daniel Gatica-Perez, 2012, “Using self-context for multimodal detection of head nods in face-to-face interactions”, in *Proceedings of the 14th ACM international conference on Multimodal interaction* (Santa Monica, California, USA), p. 289-292, 2388734: ACM.
 16. Perroud, Didier, Leonardo Angelini, Omar Abou Khaled and Elena Mugellini, 2012, “Context-Based Generation of Multimodal Feedbacks for Natural Interaction in Smart Environments”, in *AMBIENT 2012, The Second International Conference on Ambient Computing, Applications, Services and Technologies*, p. 19-25.
 17. Benoit, Alexandre, Laurent Bonnaud, Alice Caplier, Phillipe Ngo, Lionel Lawson, Daniela G. Trevisan, Vjekoslav Levacic, Céline Mancas and Guillaume Chanel, 2009, “Multimodal focus attention and stress detection and feedback in an augmented driver simulator”, *Personal and Ubiquitous Computing*, vol. 13, no 1.
 18. Palanque, Philippe, and Amélie Schyn, 2003, “A Model-Based Approach for Engineering Multimodal Interactive Systems”, in *9th IFIP TC13 Int. Conf. on Human-Computer Interaction*, IOS Press.
 19. Oviatt, S., and Seattle Incaa Designs, 2007, “Implicit user-adaptive system engagement in speech, pen and multimodal interfaces”, in *Automatic Speech Recognition & Understanding, 2007, IEEE Workshop on ASRU* (Kyoto), p. 496-501, IEEE.
 20. Henry, Tyson R., Scott E. Hudson and Gary L. Newell, 1990, “Integrating gesture and snapping into a user interface toolkit”, in *Proceedings of the 3rd annual ACM SIGGRAPH symposium on User interface software and technology* (Snowbird, Utah, USA), p. 112-122. 97938: ACM.
 21. Zaguia, Atef, Manolo Dulva Hina, Chakib Tadj and Amar Ramdane-Cherif, 2010a, “Interaction context-aware modalities and multimodal fusion for accessing web services”, *Ubiquitous Computing and Communication Journal*, vol. 5, no 4.
 22. Alexander, Christopher, S Ishikawa and M Silverstein, 1977, “Pattern languages”, *Center for Environmental Structure*, vol. 2.
 23. Zhang, Lei, and Qiang Ji, “A Bayesian network model for automatic and interactive image segmentation”, *IEEE Transactions on Image Processing*, 20.9 (2011): 2582-2593.
 24. Weber, P., Medina-Oliva, G., Simon, C., and Iung, B., (2012), Overview on Bayesian networks applications for dependability, risk analysis and maintenance areas, *Engineering Applications of Artificial Intelligence*, 25(4), 671-682.
 25. Zaguia, A., Tadj, C., and Ramdane-Cherif, A., (2015), Prototyping using a Pattern Technique and a Context-Based Bayesian Network in Multimodal Systems, *International Journal on Smart Sensing & Intelligent Systems*, 8(3).
 26. Friedman, Nir, Iftach Nachman and Dana Peér, 1999, “Learning bayesian network structure from massive datasets: the ‘sparse candidate’ algorithm”, in *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, p. 206-215, Morgan Kaufmann Publishers Inc.
 27. Jensen, Kurt, 1987, “Coloured Petri nets Petri Nets: Central Models and Their Properties”, in, edited by Brauer, W., W. Reisig and G. Rozenberg. Vol. 254, p. 248-299, Coll. “Lecture Notes in Computer Science”, Springer Berlin / Heidelberg. < <http://dx.doi.org/10.1007/BFb0046842> >
 28. CPN-Tools, 2012, “CPN-Tools”, < <http://cpntools.org/> >.