Artificial Intelligence in Medicine (2008) 42, 213—227

ARTIFICIAL INTELLIGENCE IN MEDICINE

http://www.intl.elsevierhealth.com/journals/aiim

Nutri-Educ, a nutrition software application for balancing meals, using fuzzy arithmetic and heuristic search algorithms

Jean-Christophe Buisson [a,b,*]

[a] University of Toulouse, INPT, IRIT (ENSEEIHT), 2 rue Camichel, 31071 Toulouse, France
[b] Rangueil University Hospital, Toulouse, France

Received 22 June 2007; received in revised form 10 December 2007; accepted 11 December 2007

KEYWORDS
Fuzzy arithmetic;
Heuristic search;
Nutrition;
Nutri-Educ

Summary

Objective, methods and results: Nutri-Educ is a nutrition software application developed in cooperation with the Diabetology Department of Toulouse's Rangueil University Hospital. It aims at helping any person to balance their meals. More specifically, its main goal is to enable a user to describe a meal and assess its content, and in most cases to find a small set of acceptable actions which make it well-balanced and in accordance to the user's energetic needs.

Fuzzy numbers are used to represent the inherent imprecision and fuzziness of food quantities and nutrient values as well as to model the gradual boundaries of the daily recommended values associated with each nutrient. Fuzzy arithmetic is used to perform computations on such quantities and fuzzy pattern matching provides measures of the compatibility of data to nutrient norms. Innovative visual gauges have been designed to display this information in a simple, yet comprehensive way. Finally, heuristic search algorithms are used to find a set of actions, acceptable from a nutritional point of view, which will transform the initial meal into a well-balanced one.

Conclusion: Fuzzy arithmetic proves to be an adequate model for naturally representing food quantities and values as well as for performing all necessary computation and compatibility assessments. By combining it with new interface techniques and heuristic search algorithms, it allows the Nutri-Educ software application to balance and improve meals, a complex qualitative and quantitative problem which is solved in 87% of our benchmark database.

* Correspondence address: University of Toulouse, INPT, IRIT (ENSEEIHT), 2 rue Camichel, 31071 Toulouse, France.
Tel.: +33 561 588 200; fax: +33 561 588 306.
E-mail address: buisson@enseeiht.fr.

0933-3657/$ — see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.artmed.2007.12.001

1. Introduction

Improving on the long term the dietary habits of people is a socially important task, which would help to decrease the frequency of cardio-vascular disease and the morbidity of many chronic diseases such as diabetes. Our group has been working for a number of years on computer-based educational tools for diabetics and people who are overweight [1]. Nutri-Educ is one of these tools, and possesses several unique features. First, it is the only one to deal with the inherent imprecision of nutritional data, and to use it in computations and assessments in a mathematically sound way. Very surprisingly, there is no other application software associating nutrition and fuzzy sets to our knowledge; some works [2] only describe how the concept of fuzzy set may help to define the ideal intake for foods. Secondly, Nutri-Educ is able to 'fix meals', that is to say to compute sets of actions which transform a meal to make it well balanced. Many other commercial and research tools help people build diet plans and analyze the composition of meals, but surprisingly, we did not find any others which deal with this problem of balancing.

We will first describe fuzzy arithmetic as a tool for representing and propagating imprecise data. Then we will focus on the interface methods used to present results. Finally we will describe the balancing algorithm, and how Nutri-Educ integrates all these aspects.

2. Fuzzy arithmetic and nutrition

2.1. Problem statement

In this part we are interested in using fuzzy sets theory [3] to represent and manipulate imprecise quantities. The values, which are used for the nutritional computations, are indeed affected by imprecision, and sometimes by uncertainty; it affects the weights of the manipulated foods and also their nutrient values. It also concerns the internationally recommended values which are intervals with gradual boundaries.

We first need a mathematically sound model, adapted to the representation of such values. Several potential theories exist, such as interval arithmetic [4] and classical fuzzy arithmetic [5]. Their respective advantages and disadvantages for the problems our applications need to deal with have been examined and classical fuzzy arithmetic was found to be the better adapted of the two. Probability theory was not selected as a possible solution, since it is more adapted to the representation of random uncertainty than imprecision, which is our central problem.

When such a set of representations of imprecise and/or uncertain values is stored in its memory, a computer then needs to be able to operate on these values. It should be able to do all the ordinary arithmetic on this type of values: adding, subtracting, multiplying by constants (precise or imprecise), computing mean values, etc. [4—6]. The computer will need to be able to compare imprecise values and also to be able to evaluate to which extent a value belongs to an interval with gradual boundaries. This belonging will not be a Boolean, nor a simple value nor an interval [7], but a set of possibility and necessity values.

These fuzzy computations will then be used in a generalized heuristic search algorithm, which will be able to find the smallest set of acceptable actions to transform a given meal into a well-balanced meal.

Finally, we will need interface methods for end users and nutritionists in order for them to be able to interpret the meaning and all the nuances contained in such representations, or to be able to perform a data collection task that will lead to the construction of such representations. We believe that adapted interface techniques can be an important factor for the acceptance of fuzzy models by a wider audience. Our own research work in this field [8] will be detailed here; it has been validated by several software applications, Nutri-Educ being one of them [9,10].

2.2. Sources of imprecision and uncertainty in nutrition

2.2.1. Food composition

Many food composition tables exist, both as paper documents and as computer databases. They are usually associated with a particular country, since even common foods such as yoghurts come in different sizes and have different ingredients, depending on the country. Since 1892 in the United States, the Department of Agriculture (USDA) has been keeping, in a very organized and systematic manner, a freely accessible database which contains the composition of all raw and processed foods available in the country [11,12]; more than 440,000 nutrient values are recorded for more than 6000 foods. For reasons of future compatibility, and considering the quality of their database organization, we have adopted a similar one in all our software applications, which presently contains only French food values.

There is often a problem of imprecision with the nutrient values but no problem of uncertainty: it is generally known which nutrients are present in a food, but the problem is to know in what amounts they are present. If there is doubt as to the presence of a particular nutrient, its supposed amount will be very small, which takes us back to the problem of the imprecision of a near-zero value. This problem may become more acute for industrially processed foods, for which manufacturers give only partial information. In the American USDA food values table, each quantity (a precise number) is given with two values: a measurement, and a standard deviation. The standard deviation values are unfortunately often zero, which means that the values have been obtained directly from the manufacturer, without verification from an independent laboratory.

When the food has been processed and packaged in a precise and stable way (for instance oil, sugar, biscuits, etc.), the nutrient values are often known precisely and with certainty.

For all other foods, the nutrient value for 100 g of a food referenced normally (for instance: 'apple', 'white bread', 'potatoes') may vary over a range that may be fairly large, depending on the circumstances. The carbohydrate quantity in 100 g of apple, for example, depends on the variety of apple, as well as on its degree of ripeness. Even after controlling for ripening and variety, there are variations due to soil and growing conditions.

2.2.2. Weights of foods eaten

The weights of foods absorbed by a person during a meal are often imprecisely known, as can be seen for both foods mentioned in the following extract (Fig. 1) from a food survey of a 10-year-old child.

In both cases, the food quantity is evaluated using its natural linguistic qualification: "bunch" for grapes, "piece" for cheese. After, a sizing adjective may be added to this qualification: "a small bunch of grapes", "a big piece of cheese", etc.

More exhaustively, we have tried to verify that this mode of linguistic qualification of a food quantity is the most widely used, by studying the results of food surveys done on 1607 children of ages ranging from 10 to 14, in the setting of the 'Nutri-Advice' study which was conducted for 3 years at six junior high schools in the Toulouse area.

raisin (1 petite grappe
1 petit bout de camembert

Figure 1 Extract from a food survey of a 10-year-old child. She uses naturally expressed portions ('grappe' = '-bunch', 'bout' = 'piece'), sometimes with a sizing adjective ('petit' = 'small').

We have categorized in these surveys all nominal groups which expressed a food quantity. Results are given in Fig. 2. It can be seen that the last three qualification classes are by far the largest ones.

2.2.3. Norms and international recommendations

The international medical research community periodically publishes norms and guidelines describing the characteristics of a balanced diet for healthy children and adults as well as for people suffering from chronic illnesses which require particular diet restrictions [13—15].

These norms indicate:

- The daily caloric amount related to age, height, gender, level of physical activity, and (to a lesser extent) weight. This value is the amount of energy a person must absorb everyday, not taking into account the kinds of foods to be eaten.
- The amount of particular nutrients (carbohydrates, fat, proteins, saturated fat, etc.) in this daily caloric amount. These values are often expressed as a percentage of this total (for instance, energy coming from carbohydrates should represent 50% of total energy); it is sometimes expressed as an absolute value for the day (for instance, a child under 12 needs 100 mg of calcium everyday, independent of her daily energy needs).

Precise qualification (ex: 'milk bread: 70 gr')	**11%**
Qualification with a natural portion (ex: '1 glass of milk')	**43%**
Imprecise qualification with a natural portion, modulated by 'small' or 'big' (ex: '1 small piece of camembert')	**28%**
Other kinds of imprecise qualification (ex: '1 slice of cake (1/6)', 'some potato chips, about 25', 'nutella: 1,5 teaspoons')	**18%**

Figure 2 Categorization of nominal groups qualifying food quantities.

These norms are intervals with imprecise boundaries.

2.3. Fuzzy arithmetic for the representation of imprecision and fuzziness in nutrition

In this part we will present fuzzy arithmetic and show that it is well suited for representing imperfectly known nutritional data. We will examine in particular its ability to:

- represent as directly as possible, and with all necessary nuances, the different forms of imprecision or fuzziness found in food surveys, nutritional definitions and international norms and guidelines.
- finely evaluate the compatibility of a piece of data within a category, a fundamental operation when assessing the balance of a real or hypothetical meal.

2.3.1. Fuzzy quantities

A fuzzy quantity (or fuzzy interval or fuzzy number) is a fuzzy set [3] of real numbers, written M, with a membership function μ_M which is unimodal and upper semicontinuous, that is to say such as $\forall\alpha\in]0, 1]$, $M_\alpha = \{r|\mu_M(r) \geq \alpha\}$ (the α-cut of M) is a closed interval (Fig. 3).

A fuzzy interval generalizes the concept of a closed interval, including ordinary real numbers. It can model the membership area of a variable x with more sophistication than an ordinary interval. More precisely, the support $S(M) = \{r|\mu_M(r) > 0\}$ is the largest membership area of x (x cannot take a value outside $S(M)$), whereas the kernel $\dot{M} = \{r|\mu_M(r) = 1\}$ is the set of the most plausible values for x, also called modal values.

A fuzzy number is adapted to the representation of imprecise quantities. In most situations where a parameter must be evaluated (the value of which is not known with precision), an ordinary interval is not satisfactory. If we choose a fairly large interval in order to be sure that the value is in it, the subsequent calculations based on this representation may lead to results that are not specific enough to be of real value. On the contrary, if we choose too narrow an interval, by restricting it to the most plausible values, the high precision of the results might be incorrect if an error has been made at the beginning and the actual value may not be included in the interval. A fuzzy interval allows both an optimistic and a pessimistic view: the support of the number is chosen large enough to be sure that no value is needlessly excluded with the kernel representing the most plausible values.

The most important point is to determine both the set of values which are completely impossible (the values for which $\mu(x)$ equals 0) and the set of values which are completely possible (for which $\mu(x)$ equals 1) precisely: the remaining subsets of the domain will correspond to gradual transitions.

Three different shapes for membership function (MF) are commonly used: triangular MF, trapezoidal MF and Gaussian-type MF. Triangular MF is excluded because the kernel is reduced to a singleton whereas it is often an interval in our application. Gaussian-type MF would perhaps represent fuzzy values more exactly, but their support and kernel are difficult to define precisely for dieticians for example and they lead to more intensive computations, especially when running the balancing algorithm.

A trapezoidal form for the membership function is enough to ascertain the most important properties. From a computational point of view, we will model such functions by 4-uples ($\bar{m}$, $\underline{m}$, α, β) (Fig. 4).

Fig. 5 represents several typical situations related to nutrition, with trapezoidal fuzzy numbers.

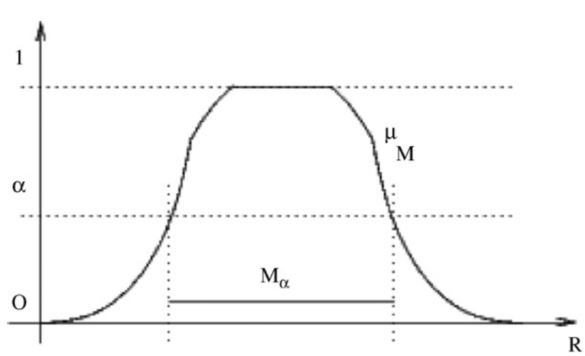

Figure 3 Fuzzy interval, shown with an α-cut M_α of height α.

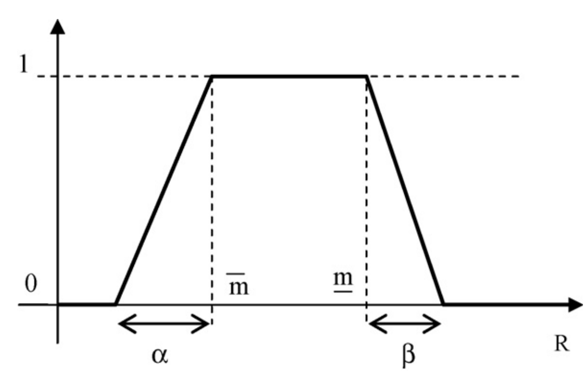

Figure 4 Fuzzy interval modeled by a 4-uple ($\bar{m}$, $\underline{m}$, α, β).

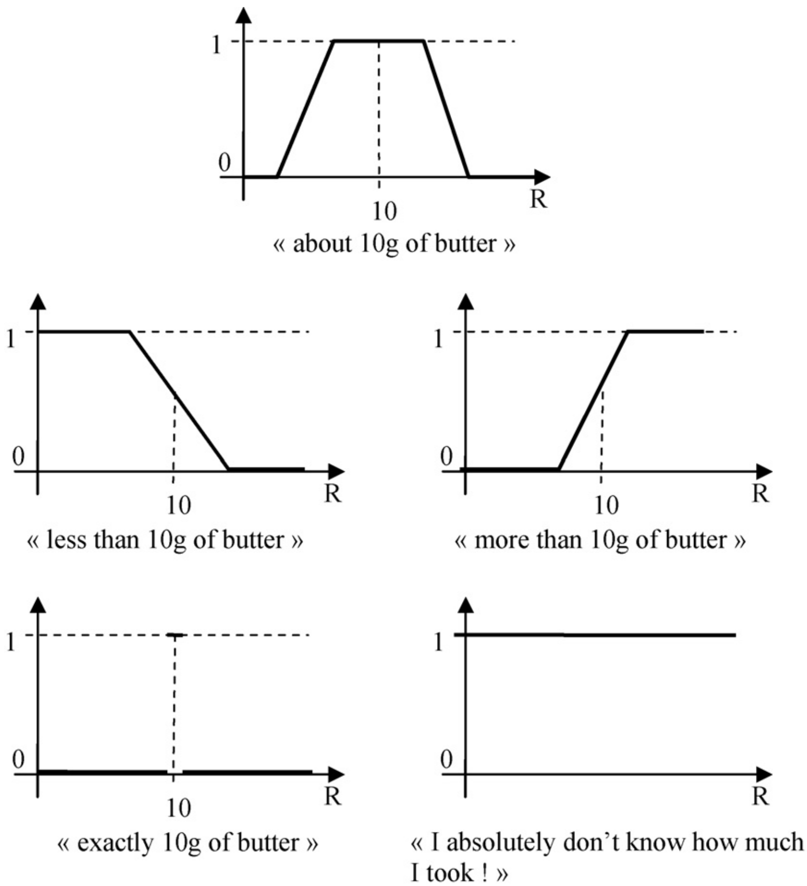

Figure 5 Typical situations of fuzzy nutritional values.

2.3.2. Implementation of fuzzy computation

Our problem here is to compute the fuzzy quantity which limits the domain of a variable $f(\omega)$, given the fuzzy quantity which limits the variable ω. Here f can be a common mathematical operation, unary or binary.

When f is continuous, it can be shown that

$$f(M,N)_\alpha = f(M_\alpha, N_\alpha) \tag{2}$$

In a word, the α-cut of the image is the image of the α-cuts and a fuzzy computation appears as interval computations made for all possible values, from 0 to 1, from a prudent computation with the supports (α = 0+), to a daring computation with the kernels (α = 1).

Eq. (2) allows us to compute $f(M_1, M_2)$ for common arithmetic operators, M_1 and M_2 being two trapezoidal fuzzy numbers ($\bar{m}_1$, $\underline{m}_1$, α_1, β_1) and ($\bar{m}_2$, $\underline{m}_2$, α_2, β_2). For addition, for example, it is easy to see that

$$(\bar{m}_1, \underline{m}_1, \alpha_1, \beta_1) + (\bar{m}_2, \underline{m}_2, \alpha_2, \beta_2)$$
$$= (\bar{m}_1 + \bar{m}_2, \underline{m}_1 + \underline{m}_2, \alpha_1 + \alpha_2, \beta_1 + \beta_2) \tag{3}$$

For multiplication and division, an approximation is necessary, since the result is no longer exactly trapezoidal. The two most important elements of the result are the support and the kernel, which can be computed precisely. The approximation consists in making the results trapezoidal, by drawing straight lines between the support and the kernel, instead of linear by part shape of the exact result. This approximation leaves the order of the membership degrees on [0, 1] unchanged.

Moreover, Schneeman has shown [16] that an approximation operator can be used, which preserves the expected interval and ordering. We intend to use this operator in a near future, provided that it does not increase computing time too much in the balancing algorithm.

2.3.3. Fuzzy pattern matching with a single variable

After having computed a fuzzy quantity, for example the total amount of energy in a meal, we need to compare it to a norm, in order to assess whether it is compatible. Let P and D be the pattern (the norm) and the data, represented by the possibility distributions μ_P and μ_D, respectively. It has been shown [17] that the following two scalar measures express several important aspects of the compatibility of data D with pattern P:

$$\Pi(P;D) = \sup_u \min(\mu_P(u), \mu_D(u)) \tag{4}$$

$$N(P;D) = 1 - \Pi(\bar{P};D) = \inf_u \max(\mu_P(u), 1 - \mu_D(u)) \tag{5}$$

$\Pi(P; D)$ evaluates to which degree it is possible that P and D refer to the same value; it is the measure of the overlapping of D and P. $N(P; D)$ measures to which degree it is impossible that D belongs to $\bar{P}$; it is the degree of inclusion of D into P. $\Pi(P; D)$ is an optimistic measure of compatibility, whereas $N(P; D)$ is pessimistic and requires verification.

Fig. 6 shows how these two values can be obtained geometrically from trapezoidal fuzzy values.

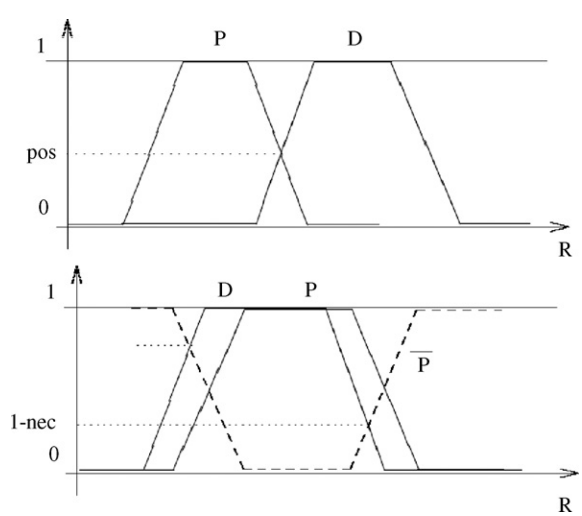

Figure 6 Geometrical determination of the possibility and necessity values of the compatibility between data D and pattern P.

2.3.4. Fuzzy pattern matching with several variables

We are now able to compute a set of compatibility values $\Pi(P_i; D_i)$ and $N(P_i; D_i)$ by separately matching all data D_1, D_{2i}, ..., etc., with the corresponding pattern P_1, P_{2i}, ..., etc. Each pair D_i, P_i represents what is known about the value of a nutrient for example, and the corresponding norm for this nutrient in the meal.

In our applications however, the nutritionists assign different degrees of importance to the nutrients considered. For example, energy, saturated fat and carbohydrates are three nutrients which have a far greater importance than others for normal adults. The degree of relative importance may vary with the group considered: calcium is more important for children than adults, cholesterol level is very important for people with cholesterol problems, etc.

Nutritionists spontaneously weigh the importance for each individual norm; for example for an adult with no medical problems, they assign 4 to energy, saturated fat and carbohydrates, 2 to protein and 1 to all other nutrients.

Let us call ω_1, ω_{2i}, ..., ω_n the respective weighting of patterns P_1, P_2, ..., P_n, with $\forall_i$, $\omega_i \leq 1$, and suppose that at least one of the degrees ω_i is 1. For example, for an adult with no medical problems, these degrees are 1 for energy, saturated fat and carbohydrates, 0.5 for protein and 0.25 for all others.

If S_i denotes the degree of compatibility (possibility or necessity) between D_i and P_i, then the global compatibility degree can be computed by

$$s = \min_{i=1,n} \max(1 - \omega_i(u), s_i) \tag{6}$$

3. Man/machine interface for a fuzzy system

Little research has been done in the field of man/machine interface for fuzzy systems, however, such interfaces are the most visible aspects of models using fuzzy set theory. Intuitive interface techniques, ergonomically adapted to an application, would be helpful assets for the acceptance of fuzzy models.

3.1. Gauges for visualizing compatibility between data and patterns

For a long time, messages such as "the amount of carbohydrates in your meal is higher than recommended" have been given to users. Yet this kind of presentation has always looked clumsy to us, and we had the feeling that this impression was shared by users, whether they were nutritionists, dieticians or patients. Moreover, such sentences take quite a long time to read and be understood and each time some aspect of the situation changes, a new sentence has to be produced and read, etc.

Then we had the idea to use an analogy with measuring devices with needles (to measure intensity, pressure, etc.), which would display both analogical values and areas of admissible values. We will call such devices 'gauges'; they display both a fuzzy quantity D and an associated norm P (Fig. 7). At a glance, we can assess the compatibility of a value with a norm. When the value or the norm changes, the nature of the change is perceived immediately on the gauge. This feature of the visual system being able to attract the attention of users allows for the

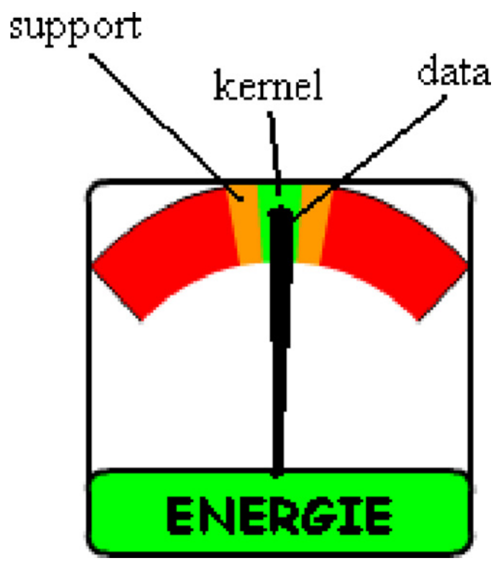

Figure 7 The different areas of a gauge. Here, the data (needle) is slightly fuzzy, but is nevertheless completely included in the norm kernel.

possibility that they could be used in monitoring systems.

The data is represented by a needle of variable width. If it is precise, the needle is thin, with a width of only one pixel. If the data is imprecise but not fuzzy, the needle is a circular arc filled in black, the two extreme borders corresponding to the interval boundaries. If the data is fuzzy (as in Fig. 7), the kernel is represented by a black filled circular arc, and the remaining part of the support is represented by thin needles every 2°, to show that the value could be one of these values.

As for the norm pattern, it is represented by colored areas under the needle. It is red outside the support (forbidden), the kernel is green (authorised) and all other areas (support—kernel) are orange. The bottom of the gauge displays a short label naming the data represented, and it is written on a background the color of which expresses the level of compatibility between the data and norm pattern. This background is green when the whole needle is included in the pattern's kernel (possibility = 1, necessity = 1), it is red when the whole needle is outside the support (possibility = 0, necessity = 0), and it is orange in all other, less clear-cut situations (possibility $>$ 0).

We can see in Fig. 8 various typical situations. In the energy example, the data is fuzzy, but corresponds perfectly to the norm pattern, which explains the green color at the bottom of the gauge. The case of precise data can be seen in the carbohydrate example, with a medium compatibility with the norm pattern (orange bottom color). In the calcium example (green band), the norm does not have an upper limit, nor the data. Nevertheless, the compatibility is perfect (green bottom color). In the saturated fat example, the situation is analogous to the calcium example, but the reverse: the norm does not have a lower limit, but is totally incompatible (red bottom color) despite the great imprecision in data (very wide needle). The last example shows completely unknown data, displayed as a '?' and a needle covering the whole domain.

The calcium example is again considered in Fig. 8. It is a situation which arises often in our software applications, when cumulating quantities such as calcium values, and where one of the values is unknown. It leads to data where we have a lower boundary value, but no upper boundary value. We can see in this example that it is still possible to get a precise evaluation of the compatibility with a norm pattern.

3.2. Textual representations for fuzzy quantities

Our nutritionists have the responsibility of providing nutrient values for each food, using custom database applications. They do so using a written language for fuzzy numbers which has been designed in coordination with them, and which has the following properties:

- it is an extension of the usual form for decimal real numbers and for some simple algebraic forms;
- it is simple to read and write; and
- it allows them to express precise values, intervals, fuzzy numbers, upper and lower bounds, possibly fuzzy.

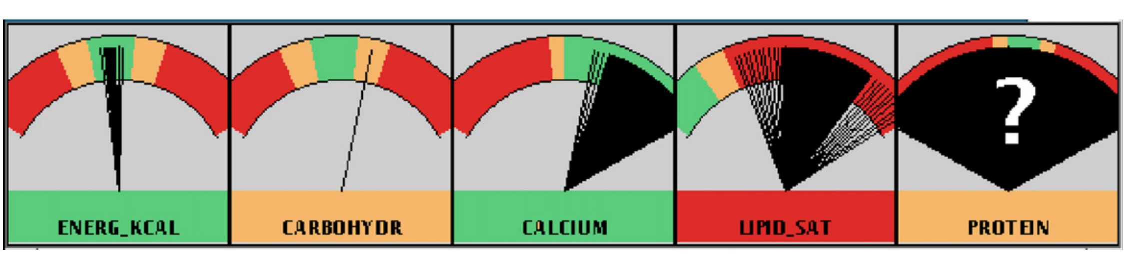

Figure 8 Different compatibility situations between several data and norm patterns.

Certainly, such a written language already exists in the form of 4-uples for trapezoidal fuzzy numbers. The nutritionists in our group are somehow able to deal with them, but it is not easy enough for them. It takes time to interpret, which is a strong flaw in applications where people must sometimes look at large arrays of data. An alternative written representation for nutritionist is therefore necessary.

As in all such written representation system, we must clearly summarize the domain of the represented objects and the domain of their representations, and how the procedures for reading and printing, which allow the conversions from one domain to another (Fig. 9), work.

Only positive fuzzy numbers are represented. We shall specify the different representation classes, by giving examples and by describing the represented numbers as 4-uples each time. These classes are

3.2.1. Precise numbers

They have the best precision at three decimal figures. For example:

$123 \rightarrow (123., 123., 0., 0.)$

3.2.2. Ordinary intervals

Their written form has been simplified, since standard mathematical notation was declared 'too difficult to write' by nutritionists. The following examples are better than a formal description:

$12.2 - 25.1 \rightarrow (12.2, 25.1, 0., 0.)$

$1000 - 1200 \rightarrow (1000., 1200., 0., 0.)$

3.2.2.1. Fuzzy numbers. They have two forms: ~val which represents val $\pm$ 10% and ~~val which represents val $\pm$ 20%. In both cases, the represented number is triangular; the support is the interval [val − val.*n*/100, val + val.*n*/100]:

$\sim 200 \rightarrow (200., 200., 20., 20.)$

$\sim\sim 3.141592 \rightarrow (3.14, 3.14, 0.628, 0.628)$

The forms ~0 and ~~0 are forbidden, since they would represent (0., 0., 0., 0.) that is to say 0., whereas a nutritionist using them would usually mean the existence of a small quantity. In any case it would be ambiguous: what difference could be made between ~0 and ~~0?

In these situations other forms, which we will describe later, can be used.

3.2.3. Intervals with fuzzy boundaries

They are represented by a combination of the previous forms for intervals and fuzzy numbers. For example:

$\sim 200 - \sim 400 \rightarrow (200.0, 400.0, 20.0, 40.0)$

$0 - \sim 23.5 \rightarrow (0., 23.5, 0., 2.35)$

Of course, the forms ~val and ~val—~val represent the same fuzzy number. The same applies to ~~val and ~~val—~~val.

3.2.4. Unbounded values

We use the characters '<' and '>', along with the forms used for fuzzy numbers. For example:

$> 400 \rightarrow (400., 0., \infty, \infty)$

$< \sim 10 \rightarrow (0., 0., 10.0, 1.0)$

To represent a value close to 0, we can use a form such as <~0.01.

In addition, we use '?' to represent the completely unknown value (0., 0., ∞, ∞).

3.2.5. Reader and printer

The reader's procedure is simple: from a written form such as ~100, a simple lexical and syntactical analysis is performed in order to reconstruct the associated 4-uple (100.0, 100.0, 10.0, 10.0).

The printer's procedure is more complicated: for a given 4-uple, it has to find the written form which represents it exactly when it exists; if it does not exist, it must find the "simplest" form which represents it "best". For example, (200.0, 400.0, 20.0, 40.0) is represented exactly by ~200—~400, but what about the 4-uple (200.0, 400.0, 20.0, 10.0)? The representation ~200—~400 does not do justice to the higher precision of the right border, and the form ~200—400 makes it appear more precise than it is in reality.

Concretely, we must try to represent the left and right borders of the value separately.

When the left border is 0 or when the right border is infinity, we will write the number '<' or '>'; when

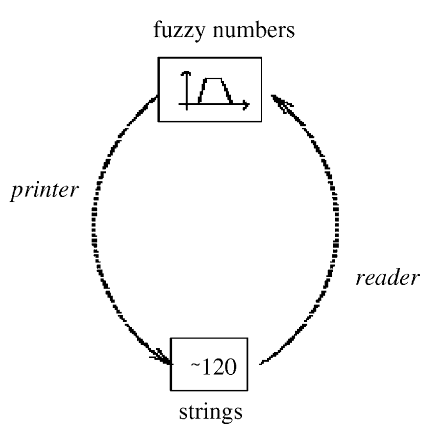

Figure 9 Reader and printer for fuzzy numbers.

the forms for the left and right borders are identical, we will write '$\sim$val' or '$\sim$val'; otherwise we will use the general interval form.

In order to find the written form for the left (resp. right) border (non zero nor infinite), we measure the proportion of the left (resp. right) spread relative to the left (resp. right) border of the kernel. When this proportion is over 20%, we use the form '$\sim$val' where val is the left (resp. right) border of the kernel; when it is over 10% we use the form '$\sim$val'; val'; otherwise (less than 5%) we use the ordinary decimal form for real numbers.

For example, from the 4-uple (3.144, 3.141, 0.314, 0.35) we can see that the left and right borders of the support are neither zero nor infinite. By studying the two numbers characterizing the left border, we see that the spread 0.314 is exactly 10% of the left border of the kernel 2.144; it is therefore represented by $\sim$3.14 (rounded to three figures). For the right border, the spread 0.35 is between 10% and 20% of the value 3.141; it is therefore represented by $\sim$3.14 (rounded to three figures). Since the forms for the left and right borders are identical, the final form for (3.144, 3.141, 0.314, 0.35) is $\sim$3.14.

3.3. Prototype pictures

We use pictures of dish portions to describe meal composition. An example of such pictures is given in Fig. 10.

All of these pictures were taken by our group of dieticians; up till now, a database of 640 pictures has been compiled, for the most common foods and dishes. We are continuing to take more pictures for less common foods.

A first problem to solve with the medical team dealt with standardizing the dish's appearance. It was decided that they would all be presented on the same ordinary white plate, with a fork and a knife on either side, in order to give a scale for the food's and dish's sizes. Of course, it could be objected that the fork and the knife should also be put into perspective since several different plates, knives and forks of different absolute sizes could give the same image. But in fact, the food on the plate will serve as a size reference for the plate, while the plate serves as a reference for the food. This circularity is only apparent since these mutual definitions will really operate only after the presentation of several different images. For example, pictures showing foods with very constrained sizes (biscuits, yoghurts, etc.) will quickly disambiguate the sizes of the plate, knife and fork for the user.

A second point to take care of was the number of different portions to photograph for each dish. Naturally, most users use three different adjectives 'small', 'medium' and 'large' when talking about the sizes of dish portions. This was confirmed in our study of the terms spontaneously used by children in food surveys. Sometimes these terms are modulated, like in 'very small' or 'fairly large', but there still remain three main categories. This partitioning created by language is self-maintaining: when we are faced with a portion picture, the fact that we classified them in the past as 'small', 'medium' or 'large' distorted our perception in order for it to be assimilated into one of these three categories, even if it requires some qualification to do it (with 'very', 'almost', etc.). Therefore language helped to form perception, and perception created language. For example, when faced with a large plate of sauerkraut, we see in addition all large plates of sauerkraut we were served in our childhood; our current perception is therefore distorted by these memories, and the perceived quantity will be an average, or rather the most typical quantity of all similar portions from our memories.

3.4. Associated possibility distributions

Therefore the three pictures 'small', 'medium' and 'large' are prototypes; they correspond to what most people think of as small, medium or large portions for themselves. Next we had to decide what possibility distributions were associated to their weights.

Since we have seen in the previous section that each of these prototype pictures represented the largest class of all analogous portions that most

Figure 10 Examples of pictures of dish portions.

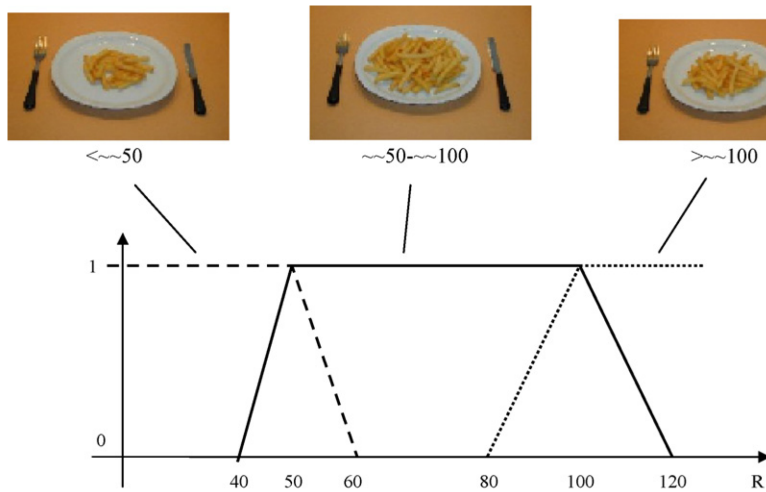

Figure 11 Weight values assigned to pictures. The possibility distributions overlap, because each picture represents the largest class of all portions to which it is assimilated.

people assimilated to it, the nutritionists had to assign large enough fuzzy weights (using the written language described previously) to them. We represented the values associated with the three pictures in Fig. 11 as examples.

3.5. Towards a fuzzy food values database

For the last 5 years the dieticians of the Nutri-Educ project have been using the description language presented in Section 3.2 to build a French food values database with fuzzy data. At the same time they are building a picture database with fuzzy quantities as described in Sections 3.3 and 3.4. When this huge work is completed, it will be interested to study how such a database improves the application software performance and user experience.

4. A heuristic search algorithm for balancing meals

4.1. Description of the problem

Nutri-Educ helps the user to build a meal which is adapted to their needs. By 'adapted to their needs', ('well-balanced' for short) we mean it conforms to international guidelines for the user, as they have been defined previously. We have also seen how to assess this global compatibility of the meal to these norms as a single necessity degree, which takes into account the individual compatibility degrees for each nutrient, and the relative importance of these individual degrees in the global assessment.

Thus, Nutri-Educ evaluates meals which belong to the state space of all possible meals which can be composed with a given set of foods or dishes. This set of components is finite: it is the set of all the foods and dishes the user chose to use, plus a small number of missing foods that the program may add to it in order to make it well-balanced. In addition, each of the foods or dishes used cannot be present in a meal in a continuous infinite quantity: they are served as one plate or half a plate in in-company or in-school cafeterias, and they are used as usual portions (spoonfuls, etc.) for foods and dishes served at home.

In fact, what links a meal to similar meals in this space is an action of addition, deletion or portion modification. These actions are both mathematical transformations on a quantity vector and nutritionally acceptable operations on a meal. Thus, this state space can now be seen as a graph, nodes being possible meals and vertices between nodes being nutritionally acceptable actions of transformation from one meal to another. In the case of fixing a meal in Nutri-Educ, the goal is to find on this graph a node which corresponds to a well-balanced meal, and which is 'as close to the initial meal as possible' (to be defined) in the graph. The path in the graph from the initial meal to the balanced meal gives the sequence of actions to apply in order to fix the initial meal and make it well balanced immediately. An example of such a graph is given on Fig. 12.

In summary, the search graph is finite, made up of meals composed of a bounded number of portions of foods or dishes. It usually is quite large, usually with thousands of nodes for a dozen food components. The goal is to find on this graph a

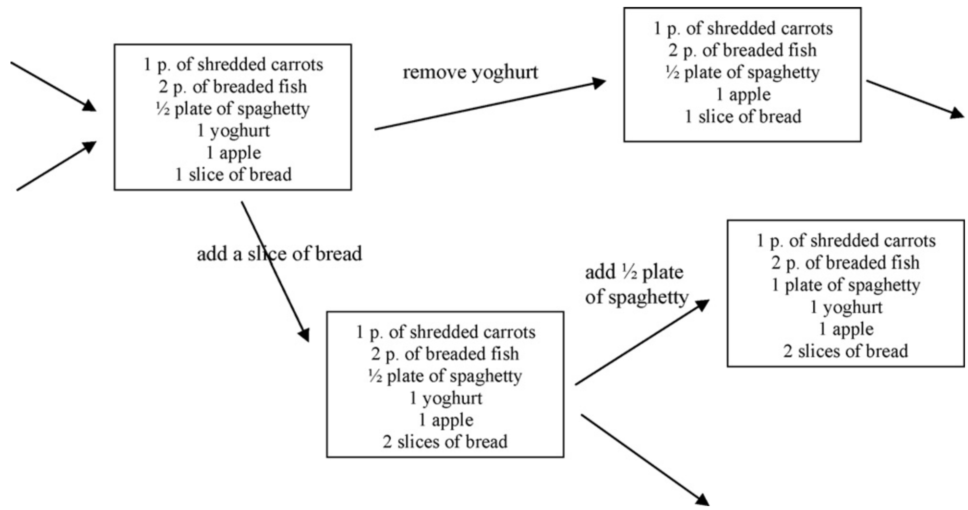

Figure 12 A small part of the graph of possible meals. Foods are used by portion, and not on a continuous scale.

well-balanced meal for the user, starting from the initial meal provided by the user. Efficient techniques must be used, since our algorithms will only have a few seconds to find the goal meal in this search space.

4.2. General principles of heuristic searches

This algorithm does not take into account the cost of an action of a change of state, which

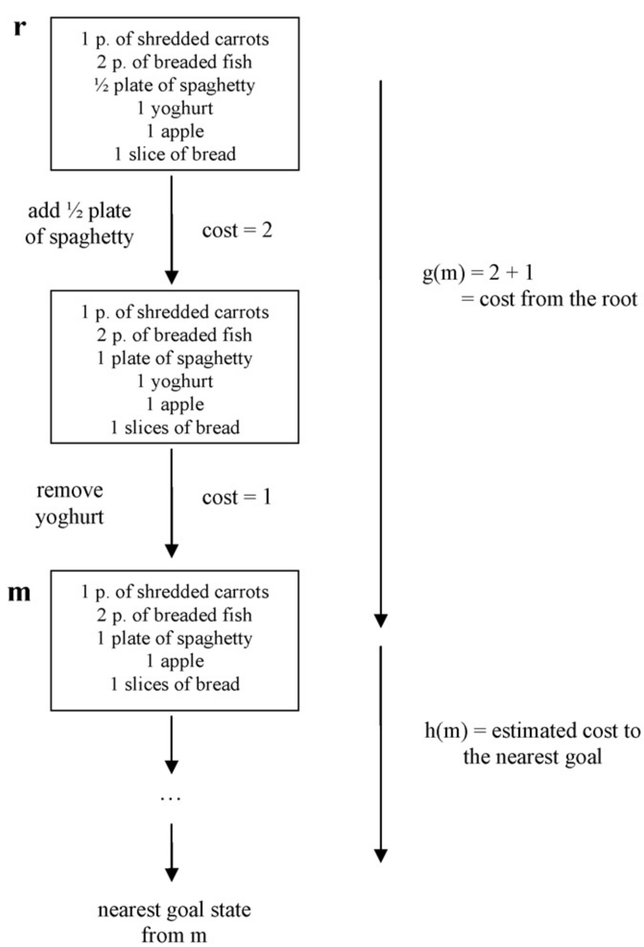

Figure 13 The *g* is the actual cost from *r* to *m*, and *h* is the estimated cost from *m* to the nearest goal state. The costs on a path are cumulated additively.

measures to which degree this change is difficult to perform, with higher costs for adding or replacing foods (since it leads to changing the recipe, doing some shopping, etc.) than for changing quantities. This minimum cost of transformation from the initial meal **r** to a meal *m* is usually noted **g**.

Now we need to use a function to estimate the remaining cost from a current state m to the nearest goal state. This term is heuristic, since if it were certain and exact, it would mean that the search results would already be known! In reality, it is an estimation that will help choose the nodes to develop. This term is usually noted **h**, and is called the heuristic term. Fig. 13 illustrates the utility of these two terms **g** and **h**.

With these two tools, we can even attempt a more ambitious goal. We can now look for, not only a goal state (a balanced meal), but also a goal state for which the cost of the path from r to it is minimal. In other words, it corresponds to finding the least expensive modifications that will make the meal well-balanced. The basic algorithm for an ordered heuristic search is as follows:

repeat
a-"generated meals" ← initial meal, "visited meals ← ∅, $g(r) \leftarrow 0$
b-focal ← meals x of "generated meals" which have the smallest $g(x) + h(x)$
c-when one of the meals of focal is well balanced, exit with success
d-choose in focal a meal m to develop
e-for each successor n of m, compute $g(n) = \min(g(n), g(m) + \mathrm{cost}(m, n))$ and put n in "generated meals" with no duplicates
f-add m to "visited meals" with no duplicates, remove m from "generated meals"
until "generated meals" = ∅; (exit with failure)

This algorithm is called A algorithm [18]. $g(m)$ is an (over)estimation of the smallest cost of the paths from r to m, and $h(m)$ is an estimation of the remaining path from m to a goal state. $f(m) = g(m) + h(m)$ is then an estimation of the cost of the shortest path from r which goes through m; by choosing at step **d**- a state which has the smallest $f(m)$, we bet that there is a state for which we estimate that it is crossed by a minimal path from r to a goal state.

The exit on failure only happens when all states of this finite graph have been explored, and none is a goal state. In practice we will never wait as long as that and we will put a limit on the number of cycles the algorithm runs. In the more general case where the graph is infinite, it can be shown [18] that such an A algorithm:

- does end up and
- always finds a goal state when one exists.

However, the optimality of this goal state is not guaranteed. If h is a minorant estimation of the remaining cost to the nearest solution, the algorithm is called A*, and it can be shown [19] that the first solution found has an optimal cost. Moreover, if we keep running the algorithm after having found the first solution, it will keep producing solutions in increasing order of cost. This feature is interesting in practice in a program such as Nutri-Educ which will provide the user with a whole set of solutions, presenting the best ones first.

The exact description of the functions g and h we used for Nutri-Educ and how fuzzy computations were dealt with in the algorithm is very technical and will not be described here; it can be found in [20]. We used a test database of 3479 real meals in order to assess the quality of various solutions. Our best algorithm, which is used in the current version of Nutri-Educ, is able to balance 3059 of the database meals in less than 20,000 cycles.

5. The Nutri-Educ application software

Nutri-Educ is a free software application which is presently available only in French and accessible at http://nutrieduc.fr (accessed: 29 November 2007); it is not open-source. The Diabetology Department of Rangueil University Hospital in Toulouse prescribes it to its diabetic patients, as part of their treatment and educational training. The software is composed of two parts: a client interface written in Flash/actionscript, and a computation server written in Java. These two parts communicate using the XML-RPC protocol. This organization allows the program to run as a desktop application or as an Internet application. The client interface was designed with large widgets in order to also be used on kiosks with touch screens. Fig. 14 presents a block diagram of the various phases of the execution of Nutri-Educ.

The 'identification' and 'user file editing' phases provide Nutri-Educ with the necessary data about the user, such as their energy needs and information on possible medical problems related to nutrition. The 'daily meal breakdown' phase allows the program to compute an energetic goal for a particular meal (breakfast, snack, lunch or dinner). All other modules are related to the meal description and its balancing assessment and improvement.

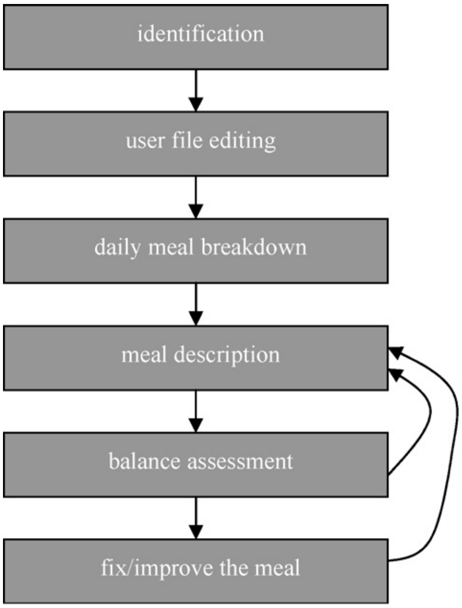

Figure 14 Block diagram of Nutri-Educ architecture.

The previous version of this application was called 'Nutri-Expert' and used the same A* algorithm to improve meal balance. 'Nutri-Educ' is a more sophisticated version where the interface has been completely redesigned to take into account fuzziness in a visual way, from food portions when querying quantities to meal corrections which are actually drawn on the initial meal description, as can be seen in Fig. 17. Ref. [20] provides a very detailed account of the A* algorithm whereas the present paper gives a more general presentation of the application and puts more emphasis on data elicitation and related medical issues.

5.1. Meal description

Fig. 15 shows how a user chooses between several food pictures. The weight of each one is represented by a possibility distribution which was given by a dietician.

5.2. Fuzzy computation and assessment

Fig. 16 shows a meal which has been provided by the user, and how it corresponds to the norms regarding four nutrients (energy, carbohydrates, fat, proteins). The score (7/10) is the combination of the compatibility measures associated with each nutrient, weighted by their importance.

5.3. Meal balancing

The user may then press the 'balance' ('correction') button, which runs the meal balancing

Figure 15 Choice between several portions of bread. Each one is associated with a weight possibility distribution.

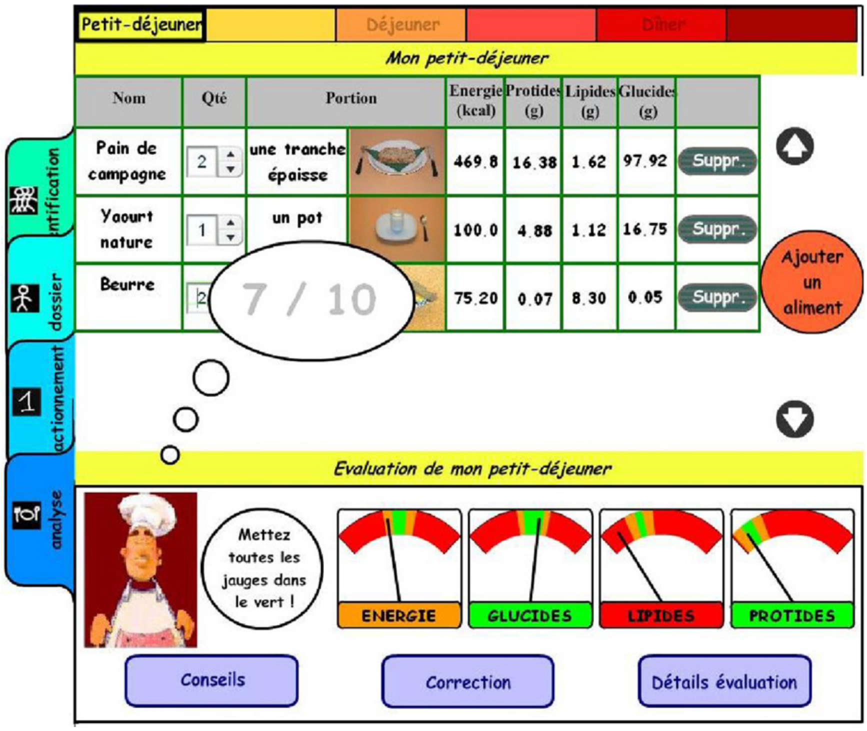

Figure 16 Fuzzy computation and norm assessment with gauges.

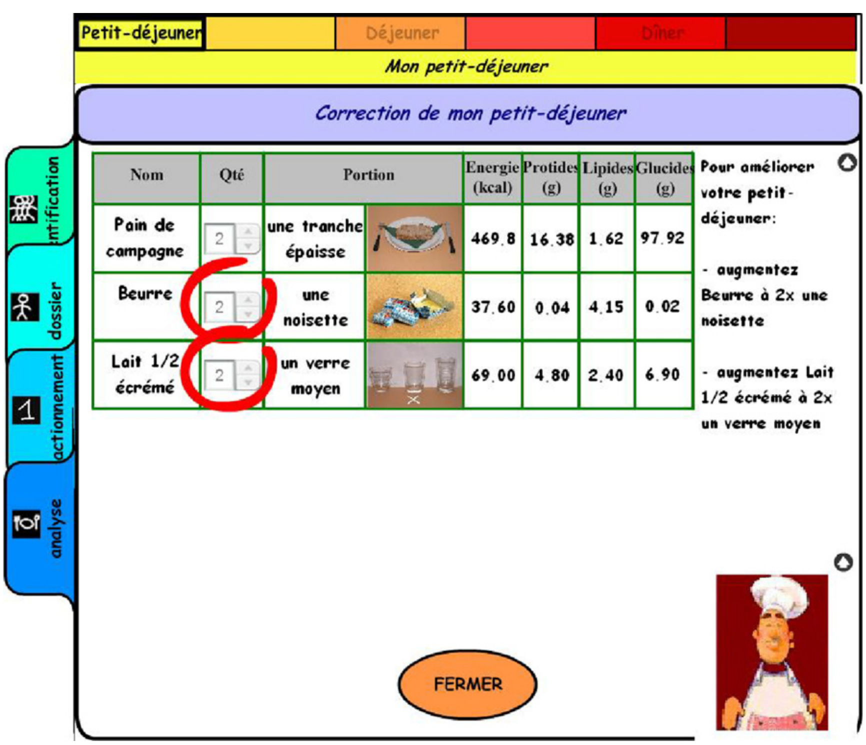

Figure 17 Meal balancing.

A* algorithm. The recommended solutions are presented both in text (right column) and visually, using food pictures as much as possible (Fig. 17).

6. Conclusion

Fuzzy arithmetic is well suited to deal with the inherent imprecision of data associated with food weights and nutrient values, and to propagate it through computations in a mathematically sound way. Fuzzy pattern matching allows the compatibility of data to a norm to be assessed and all users can understand the results with our extended gauges. Finally, a heuristic search algorithm permits the user to transform a meal to make it well balanced, a complex qualitative and quantitative problem which is successfully solved in 87% of cases.

For the future, we intend to deal with all the meals of a day and try to balance them globally instead of balancing them individually, which would make better sense from a nutritional point of view (at least for users who are not diabetics). It would greatly enlarge the state space of the heuristic search algorithm, but would also reduce the constraints on the problem and facilitate its solution. We also need to make medical evaluation studies to assert to which extent using Nutri-Educ improves the user's knowledge and modifies their daily cooking habits. If proved, its efficacy would make Nutri-Educ an easy to use and powerful tool for helping to fight the worldwide obesity problem.

References

[1] Buisson J-C. Approximate reasoning in computer-aided medical decision systems. In: Zimmermann H-J, editor. The handbooks of fuzzy sets practical applications of fuzzy technologies, 7. Boston: Kluwer; 1999. p. 337—61.
[2] Wirsam B, Hahn A. Fuzzy methods in nutrition planning and education in clinical nutrition. In: Theodorescu, Jain, Kandel, editors. Soft computing in human-related sciences. Boca Raton: CRC Press; 1999. p. 335—50.
[3] Zadeh LA. Fuzzy sets. Information and Control 1965;8:338—53.
[4] Moore RE. Interval analysis. New York: Prentice Hall; 1966.
[5] Dubois D, Prade H. Théorie des Possibilités—Application à la représentation de connaissances en informatique. Paris: Masson; 1988.
[6] Terano T, Asai K, Sugeno M, editors. M.S. applied fuzzy systems. Cambridge: AP Professional; 1995.
[7] Dubois D, Prade H. Fuzzy numbers: an overview. In: Bezdek J, editor. Analysis of fuzzy information. Boca Raton: CRC Press; 1987. p. 3—39.
[8] Buisson J-C. Knowledge development expert systems and their application in nutrition. In: Leondes C, editor. Knowledge based systems—techniques and applications. New York: Academic Press; 2000. p. 37—65.
[9] Turnin M-C, Beddok R, Clottes J, Abadie R, Martini P, Buisson J-C, et al. Telematic expert system diabeto: a new tool for diet self monitoring for diabetic patients. Diabetes Care 1992;15(2):204—12.
[10] Turnin MC, Bourgeois O, Cathelineau G, Leguerrier A, Halimi S, Sandre-Banon D, et al. Multicenter randomized evaluation of a nutritional educational software in obese patients. Diabetes Metab 2001;27:139—47.
[11] USDA-SR16. National nutrient database for standard reference, release 16. Technical report. Washington: United States Department of Agriculture USDA Nutrient Data Laboratory Home Page. 2003, http://www.ars.usda.gov/ba/bhnrc/ndl (accessed: December 1, 2007).
[12] Pehrsson PR, Haytowitz DB, Holden JM, Perry CR, Beckler DG. USDA's national food and nutrient analysis program: food sampling. Journal of Food Computing Analysis 2000;13:379—89.
[13] USDA. United States Department of Agriculture, dietary guidelines for Americans. Washington: USDA, Nutrient Data Laboratory Home Page. 2005, http://www.ars.usda.gov/ba/bhnrc/ndl (accessed: December 1, 2007).
[14] Krauss RM, Eckel R, Howard B. American heart association dietary guidelines. Circulation 2000;102:2284—99.
[15] Schneeman BO. Evolution of dietary guidelines. Journal of the American Dietetic Association 2003;103(12):5—9.
[16] Schneeman BO. Trapezoidal approximations of fuzzy numbers—revisited. Fuzzy Sets and Systems 2007;158:757—68.
[17] Cayrol M, Farreny H, Prade H. Fuzzy pattern matching. Kybernetes 1982;11:103—16.
[18] Farreny H. Recherche heuristiquement ordonnée dans les graphes d'états. Paris: Masson; 1995.
[19] Hart PE, Nilsson NJ, Raphael B. A formal basis for the heuristic determination of minimum cost paths. IEEE Transactions on Systems Science and Cybernetics 1968;4:100—7.
[20] Buisson J-C, Garel A. Balancing meals using fuzzy arithmetic and heuristic search algorithms. IEEE Transactions on Fuzzy Systems 2003;11:68—78.