Int. J. Appl. Math. Comput. Sci., 2002, Vol.12, No.3, 383–390

IMPOSING RESTRICTIONS ON DENSITY FUNCTIONS UTILISED IN COMPUTING WITH WORDS

MARCUS GEMEINDER*

* Faculty of Electrical Engineering, Chair of Computer Engineering
Fernuniversität Hagen, 58084 Hagen, Germany
e-mail: marcus.gemeinder@fernuni-hagen.de

Applying the generalised extension principle within the area of Computing with Words typically leads to complex maximisation problems. If distributed quantities—such as, e.g., size distributions within human populations—are considered, density functions representing these distributions become involved. Very often the optimising density functions do not resemble those found in nature; for instance, an optimising density function could consist of two single Dirac pulses positioned near the opposite bounds of the interval limiting the possible values of the quantity considered. Therefore, in this article, density functions with certain shapes which enable us to overcome this lack of resemblance are considered. Furthermore, some considerations on solving the resulting maximisation problems are reported.

Keywords: computing with words, approximate reasoning, generalised extension principle, resemblance

1. Introduction

A generic problem in Computing with Words (CW) is the following: From a given set of propositions expressed in a natural language (NL), an answer to a query expressed in the NL is to be inferred.

Set of propositions in NL *Initial Data Set* (IDS)	infer $\Longrightarrow$	Answer to query q in NL *Terminal Data Set* (TDS)

The meanings of the given propositions can be expressed as generalised constraints on a variable X: $X\ isr\ R$, with R the constraining relation and isr a copula, in which r indicates the type of the constraint (possibilistic, probabilistic, etc.). In the remainder of this paper, mainly constraints of the possibilistic type will be of interest.[1]

One major task in CW is, after translating a set of given propositions into constraints, to derive from these constraints consequent constraints applying the rules of constraint propagation. These rules coincide with the rules of inference in fuzzy logic. Thus, the principal constraint propagation rule is the *generalised extension principle* (Zadeh, 1999).

A typical example is the following: The IDS consists of the proposition p:*"Most Swedes are tall"*, and the query is q:*"What is the average height of Swedes?"* (Zadeh, 1999). The proposition p translates to (Zadeh, 1979)

$$\pi_p(\rho) = \mu_{\mathrm{most}}\Big(\int_{h_{\min}}^{h_{\max}} \rho(h)\mu_{\mathrm{tall}}(h)\,\mathrm{d}h\Big), \tag{1}$$

in which $\rho(h)$ is a density function defined on the interval $[h_{\min}, h_{\max}]$, $\rho(h)\,\mathrm{d}h$ is the proportion of Swedes whose height is in the interval $[h, h+\mathrm{d}h]$, and π_p is the possibility distribution of ρ induced by p. The average height of all the Swedes following from this density function is

$$\nu = \int_{h_{\min}}^{h_{\max}} \rho(h)h\,\mathrm{d}h. \tag{2}$$

The application of the generalised extension principle yields a membership function which indicates the possible average heights. This membership function is determined by solving the following maximisation problem:

$$\mu_{h_{\mathrm{ave}}}(\nu) = \sup_{\rho} \pi_p(\rho), \tag{3}$$

subject to (2).

In general—as one can easily see—the maximisation problem to be solved has the following structure:

$$\mu_q(\nu) = \sup_{\rho} \pi_p(\rho), \tag{4}$$

subject to

$$\nu = q(\rho). \tag{5}$$

[1] In this case, isd reduces to is, and R is a fuzzy relation that constrains X by playing the role of the possibility distribution of X.

In the following section, a Genetic Algorithm (GA) to solve this special kind of maximisation problem is presented. A density function $\rho(u)$[2] maximising (4) and fulfilling the auxiliary condition (5) is sought. The algorithm has to be executed several times for different values of ν. The result of each execution is a point $(\nu, \mu_q(\nu))$. These points form together a pointwise representation of the membership function $\mu_q(\nu)$.

Handling the auxiliary condition (5) is not an easy task. Therefore, in Section 3 it is described how to convert the optimisation problem (4), (5) into a multi-objective form. Special attention is paid to combining the single objective function values in order to obtain an overall ranking of potential solutions.

An aspect of special importance within the framework of CW is treated in Section 4, viz., the optimising density functions very often do not resemble those found in nature. This could narrow the usability of the results obtained when applying the methods of CW. For this reason, it is suggested to consider whether—according to the reasoning problem given—imposing restrictions on the shape of the density functions utilised could yield more useful results.

2. GA-Based Problem Solving

In (Gemeinder, 2000), a GA to solve the afore-mentioned maximisation problem is presented. Some properties of the utilised GA will be described in the sequel.

2.1. Representation of Solutions

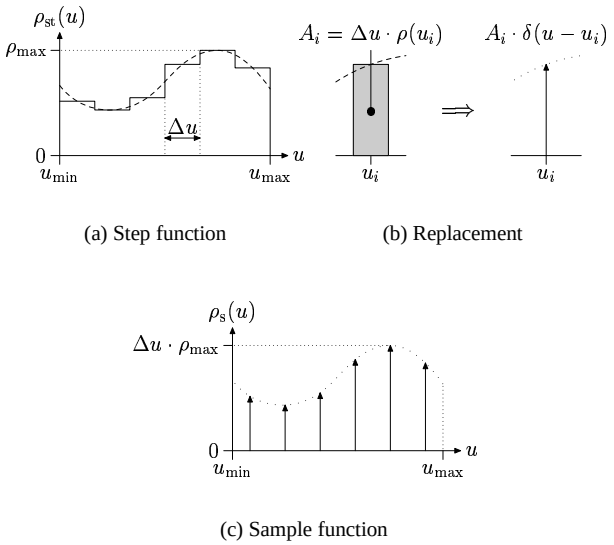
The continuous density function $n_\rho(u)$, defined on the interval $[u_{\min}, u_{\max}]$, is replaced by a discrete function consisting of n_ρ equidistant Dirac pulses as follows. Firstly, the function is approximated by a step function with n_ρ steps of equal width $\Delta u = (u_{\max} - u_{\min})/n\rho$ (see Fig. 1(a)). The function value of each step is determined by the value $\rho(u_i)$, with u_i lying in the middle of the interval covered by the respective step. And secondly, a Dirac pulse is positioned at each u_i. Each pulse is weighted by a factor w_i equal to the area of the rectangle under the corresponding part of the step function (see Fig. 1(b)). In combination, the Dirac pulses form a sample ρ_s of the density function ρ (see Fig. 1(c)).[3]

Now, each individual is a vector of n_ρ real weights w_i.

2.2. Genetic Operators

The GA utilises the widely employed standard two-point crossover for recombination. The operation of mutation is more complicated: to explore new regions of the solution space, it is sometimes useful to have large mutations, but for convergence to an optimum small changes are needed. Consequently, a mutation operator which prefers small variations but does not suppress large mutation steps (Pohlheim, 1999) is implemented: a value out of an interval $[-r, r]$ is added to the selected variable w_i. Here r is called the *mutation range*. This value is determined by a factor $a_i = 2^{-x \cdot k}$, with x being each time randomly chosen from the interval $[0, 1]$; the *mutation precision* k globally influences the size of mutation changes. The '$+$' or '$-$' sign is chosen with probability 0.5 each:

$$w_i^{\mathrm{mut}} = w_i \pm r \cdot a_i \text{ with } a_i = 2^{-x \cdot k} \text{ and } x \in [0, 1]. \quad (6)$$



(a) Step function (b) Replacement

(c) Sample function

Fig. 1. Sampling of the density function $\rho(u)$.

2.3. Fitness Evaluation and Selection

For fitness assignment linear ranking is employed. All individuals are ordered with respect to their objective function (values the possibility of ρ given p). Then the fitness Φ depends on the rank of an individual within this sorted list only (Pohlheim, 1999). For selection, *stochastic universal sampling* (Baker, 1987) is utilised.

2.4. Handling of Auxiliary Conditions

The considerations made so far are independent of two auxiliary conditions: first, the constraining condition (5), and second, a general condition to be fulfilled by a density function: the integral $\int_{-\infty}^{\infty} \rho(u)\,\mathrm{d}u$ has to have the value 1. This second condition is easily fulfilled by normalising the individuals after applying the genetic operators.

[2] Here u denotes an element of U, the universe of discourse.

[3] Here δ is the Dirac distribution.

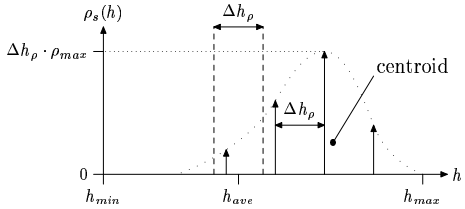
Considering the first condition is more complicated. There are two principal methods to choose from: the first one is to punish an offence against such a condition by lowering the corresponding fitness; the second one is to prevent such an offence.

In the application described in (Gemeinder, 2000), the second way was chosen. Special operators which guarantee that the auxiliary conditions are fulfilled have to be developed. In the case of the example above, an operator which shifts the sample functions along the h-axis in steps of width Δh_ρ until the average height ν_{act} lies in the interval $(\nu_m - \frac{1}{2}\Delta h_\rho, \nu_m + \frac{1}{2}\Delta h_\rho]$ (see Fig. 2) has been implemented:[4]

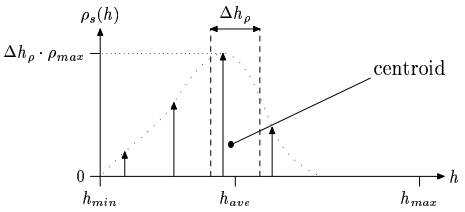
$$\rho_s^{\mathrm{sh}}(h) = \rho_s(h + k\cdot\Delta h_\rho) \quad \text{with} \quad k \in \{0, \pm 1, \pm 2, \ldots\} \tag{7}$$

and

$$\nu_{\mathrm{act}}\left(\rho_s^{\mathrm{sh}}(h)\right) = \int_{h_{\min}}^{h_{\max}} \rho_s(h + k\cdot\Delta h_\rho) h \,\mathrm{d}h \in (\nu_m - \frac{1}{2}\Delta h_\rho, \nu_m + \frac{1}{2}\Delta h_\rho]. \tag{8}$$



(a)



(b)

Fig. 2. Shifting of the density function $\rho(h)$ to fulfill the auxiliary condition: (a) density function after applying the genetic operators; (b) resulting density function after two shifting steps.

This does not lead to a loss of accuracy, because the optimum fitness value for each ν_m is reported together with the corresponding actual average height $\nu_{\mathrm{act},m}$. So we obtain n_ν accurate points $(\nu_{\mathrm{act},m}, \mu_{h_{\mathrm{ave}}}(\nu_{\mathrm{act},m}))$ representing the linguistic value h_{ave}.

The main advantage of the prevention method is that the resulting performance of the algorithm is very good. But there is obviously a great disadvantage, too: for each class of problems a special operator has to be implemented. The operator mentioned above can be used to solve the example problem, but if the query was, e.g., q: *"How many Swedes are small?"*, another operator would be needed.

Owing to this disadvantage, another approach as described in the sequel was investigated.

3. Multi-Objective Problem Modelling

Another example—the so-called Robert example (Zadeh, 2001a)—will serve to illustrate the prevention method's limits of applicability. In its initial form, based on the proposition p: *"Usually Robert returns from work at about 6 p.m."* an answer to the query q: *"What is the probability that Robert is at home at 6:30 p.m.?"* is to be inferred. This leads to an optimisation problem with similar complexity as in the case of the Swedes example, which could be handled in a similar way as described before.

Considering Version 2 of the Robert example, we obtain a more complex optimisation problem: instead of only one proposition, here the IDS consists of two propositions p_0:*"Usually Robert leaves office at about 5:30 p.m."* and p_1:*"Usually it takes him about 30 minutes to drive home"*; the query q is not changed. Let $g_0(u)$ and $g_1(u)$ be probability density functions, indicating the distributions of T_0, the time at which Robert leaves his office, and the travel time T_1, respectively.

The generalised constraints induced by the IDS are of the type $X\ isu\ R$, which is an abbreviation for *usually* $X\ is\ R$. This term leads to the expression

$$P(X \text{ is } R) \text{ is } \textit{usually}, \tag{9}$$

where $X\ is\ R$ is a fuzzy event and *usually* its fuzzy probability, which is the possibility distribution of its crisp probability (Zadeh, 1999).

Therefore, after converting the two propositions in this way, one obtains the two possibility distributions (Zadeh, 2001b)[5]

$$\pi_{p_0}(g_0) = \mu_{\mathrm{usually}}\Big(\int_0^{12} g_0(u)\mu_{5:30^*}(u)\,\mathrm{d}u\Big), \tag{10}$$

$$\pi_{p_1}(g_1) = \mu_{\mathrm{usually}}\Big(\int_0^{12} g_1(u)\mu_{0:30^*}(u)\,\mathrm{d}u\Big). \tag{11}$$

[4] Here ν_m denotes the average height considered currently. The index 'sh' stands for *shifted*.

[5] The asterisk stands for *about*.

For the time when Robert arrives at home we have $T = T_0 + T_1$. Therefore, the TDS translates into the following constraint (Zadeh, 2001b):

$$\nu = q(g_0, g_1) = \int_0^{6:30} \Big(\int_0^{12} g_0(w)\, g_1(u-w)\, \mathrm{d}w \Big)\, \mathrm{d}u. \quad (12)$$

Now, one has to solve the maximisation task

$$\mu_q(\nu) = \sup_{g_0, g_1} \Big(\pi_{p_0}(g_0) \wedge \pi_{p_1}(g_1) \Big) \quad (13)$$

subject to (12). It is obvious that there is no straightforward way to adjust the density functions g_0 and g_1 after applying the genetic operators in a way that avoids offences against the auxiliary condition. For this reason, in (Gemeinder, 2001b) another way of modelling the optimisation problem was proposed.

Adopting this proposal, the constraining condition (5) is modelled as a further objective function. The optimisation problem (4), (5) is converted into

$$\text{maximise} \quad \mu_q(\nu_{\mathrm{act}}) = \pi_p(\rho), \quad (14)$$

$$\text{minimise} \quad d(\nu_{\mathrm{act}}, \nu) = |\nu_{\mathrm{act}} - \nu|, \quad (15)$$

$$\text{subject to} \quad \nu_{\mathrm{act}} = q(\rho). \quad (16)$$

Now, instead of one objective there are two objectives, whereby the second one—in conjunction with (16)—ensures, by lowering the fitness values, that the constraining condition (5) is observed. Thereby, the label 'act' indicates the actual value of the constraining condition in contrast to the desired value ν.

The main advantage of this multi-objective problem modelling is that the necessity to handle the auxiliary condition induced by the TDS explicitly vanishes. In fact, the auxiliary condition is observed implicitly by considering the second objective. There is no need any more to implement special operators which prevent offences against the auxiliary conditions when the genetic pool is evolved.

To have both representations of the maximisation problem absolutely consistent, for the second objective it has to be demanded that the value 0 is reached. But such an absolute consistence is not necessary. Anyway, as has been mentioned above, the algorithm has to be executed several times for different values of ν. Altogether, the executions form a pointwise representation of μ_q.

Therefore, in the multi-objective case $\nu_{\mathrm{act}} \approx \nu$ is sufficient if—and this is the crucial point—for this value ν_{act} the maximum possible value for the first objective is found. If so, the pair consisting of ν_{act} and the corresponding possibility $\pi_p(\rho)$ is part of the pointwise representation of the membership function μ_q without any loss of accuracy.

Nevertheless, employing the multi-objective problem formulation results in a new problem to be solved: to obtain an overall quality measure it is necessary to combine the single objective function values (14) and (15). This topic will be dealt with in the sequel.

3.1. Fitness Assignment

Several strategies for fitness assignment utilised in multi-objective evolutionary algorithms have been investigated with respect to their applicability to the optimisation problem considered (Gemeinder, 2001b). It has turned out that common approaches like computing a weighted sum dependent on the two objectives are not very suitable for this application.

Therefore, an evaluation method (called *grouping*) making substantial use of the fact that $\nu_{\mathrm{act}} \approx \nu$ is sufficient has been introduced. In a first step, the solution to be evaluated is classified by determining its membership of a certain group of solutions. The groups are assembled depending on the distance between ν_{act} and ν. Thereby, the range of distances is sub-divided into intervals of equal width Δd (see Fig. 3), and all the solutions whose distances $d(\nu_{\mathrm{act}}, \nu)$ lie in the same interval form a group. Only if the solutions to be compared are members of the same group the objective (14) is decisive.

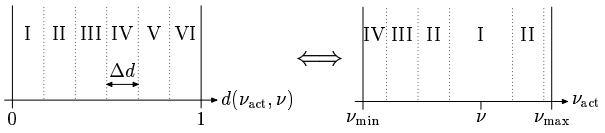


Fig. 3. Grouping of distances.

The entire procedure can be performed by calculating a single quality measure: the objective function value $f(\rho)$ characterising the solution's quality is set to[6]

$$f(\rho) = \pi_p(\rho) - \left[\frac{d\,(\nu_{\mathrm{act}}(\rho), \nu)}{\Delta d} \right]. \quad (17)$$

If the distance between ν_{act} and ν is small enough—where *small enough* is specified by the parameter Δd—only the first objective is decisive. For each combination of ν and Δd, the algorithm converges to a certain ν_{act}, and in a perfect manner to an accurate value for the membership function $\mu_q(\nu)$.

In the remainder of this article the considerations will focus on the question whether or not applying the generalised extension principle in its general form is satisfactory in any case.

[6] Here $[x]$ is the largest integer no greater than x.

4. Restricted Density Functions

Applying the approach described in the two preceding sections to the Swedes example yields (within a wide range of average heights ν) optimum possibility values which rest upon density functions similar to the one depicted in Fig. 4.

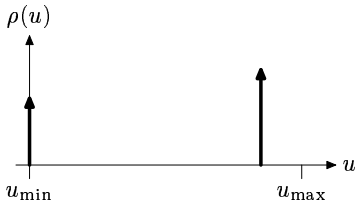


Fig. 4. Maximising density function $\rho(u)$ typical for the Swedes example.

Such density functions do not resemble those found in nature; as well as density functions typical for technical applications—such as, for instance, those describing the distribution of measures in manufacturing—have other shapes. Therefore, the values obtained by employing the theoretical framework of CW may not fit reality. This factor could narrow the practical usability of the obtained results.

Therefore, one should determine, depending on the reasoning problem under consideration, whether utilising density functions with restricted shapes instead of arbitrarily shaped ones would possibly yield more usable results.

This guiding idea was the main motivation to extend the application existing so far in a way which allows us to impose restrictions on the shapes of the utilised density functions (Gemeinder, 2001a).

Several types of shapes for the density functions are provided. The functions determining the shapes are defined on the unit interval. Therefore, the values of the quantities considered have to be mapped onto this interval. The first provided shape is a triangular one, as depicted in Fig. 5(a). For the three parameters a_1, a_2 and a_3 we have

$$a_1 < a_2 - \epsilon, \quad 0 \le a_2 \le 1, \quad a_2 + \epsilon < a_3, \tag{18}$$

with ϵ being a constant parameter which can be employed to guarantee a certain width of the support[7] of $\rho(u)$. From (18), parts of the triangle may lie outside the unit interval (see Fig. 5(b)).

Other provided shapes are depicted in Fig. 6.[8] The second shape is a trapezoidal one with parameters defined

[7] *Support* is to be understood analogously to the support of a fuzzy set.

[8] Naturally, it is easily possible to implement further ones.

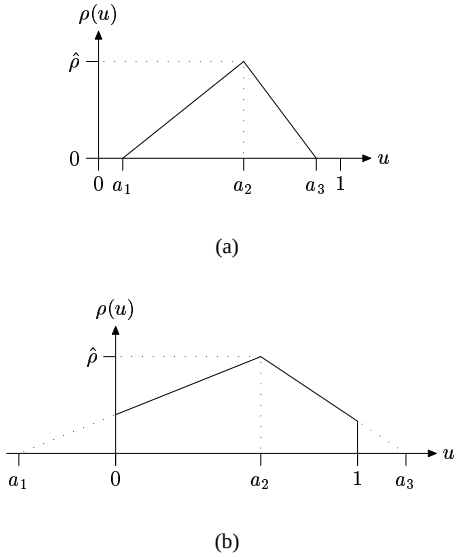


Fig. 5. Triangular shape.

in correspondence with the ones determining the triangular shape. The next one has the form of a double S-function: $\rho(u) = \hat{\rho} \cdot SS(u, a_1, a_2, a_3)$ with

$$SS(u, a_1, a_2, a_3) = \begin{cases} S(u, a_1, \dfrac{a_1 + a_2}{2}, a_2) & \text{if } u \le a_2, \\ 1 - S(u, a_2, \dfrac{a_2 + a_3}{2}, a_3) & \text{otherwise} \end{cases} \tag{19}$$

and

$$S(u, a, b, c) = \begin{cases} 0 & \text{if} \quad u \le a, \\ 2\left(\dfrac{u-a}{c-a}\right)^2 & \text{if} \quad a < u \le b, \\ 1 - 2\left(\dfrac{u-c}{c-a}\right)^2 & \text{if} \quad b < u \le c, \\ 1 & \text{if} \quad c < u, \end{cases} \tag{20}$$

with

$$b = \frac{a + c}{2}.$$

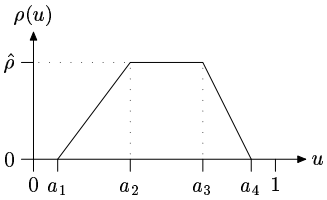
For the parameters a_i conditions (18) hold, too. The last shape has the form of a Gaussian (or *normal*) distribution:

$$\rho(u) = \hat{\rho} \cdot \exp\left(-\frac{(u - a_1)^2}{2a_2^2}\right), \tag{21}$$

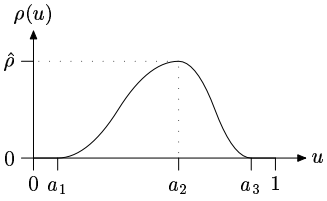
with $a_1 \in [0, 1]$, $a_2 \in \mathbb{R}^+$, and points of inflection at $a_1 \pm a_2$.

At any rate, the parameter $\hat{\rho}$ is determined by the parameters a_i: for a density function we must have

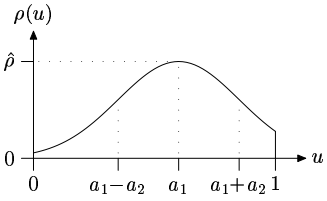
$$\int_{-\infty}^{\infty} \rho(u)\, \mathrm{d}u = 1. \tag{22}$$



(a) Trapezoid



(b) Double-S



(c) Gaussian

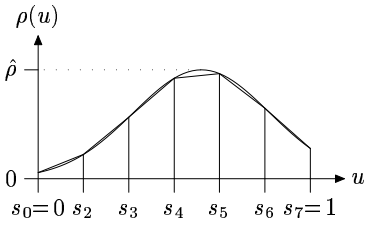
Fig. 6. Further shapes for density functions.

Therefore, the parameter $\hat{\rho}$ is calculated as

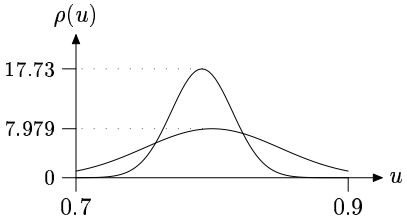
$$\hat{\rho} = \frac{1}{\int_0^1 \tilde{\rho}_1(u)\,\mathrm{d}u}, \tag{23}$$

where $\tilde{\rho}_1$ denotes one of the restricted functions normalised by a preliminary value $\hat{\rho}_1 = 1$. The necessary integration is performed numerically, employing the trapezoidal version of the closed Newton-Côtes rule (Pozrikidis, 1998). The values of $\rho_1(u)$ are computed at $N+1$ base points evenly distributed in the interval $[0, 1]$. The first base point is 0, and the last one is 1. Therefore the uniform spacing between two adjacent base points is equal to $1/N$. Now, the area below the function is approximated by the area below the polygonal line determined by the computed points (see Fig. 7(a)).

The number of base points N can be chosen freely. Table 1 provides a clue for a good choice. Regarding the example above with the interval of interest (the possible heights) set to $[100, 200]$, $\mu_{\mathrm{tall}} = S(h, 165, 175, 185)$ and $\mu_{\mathrm{most}} = S(h, 0.5, 0.7, 0.9)$, the possibility (1) was computed for different values of N, considering the two density functions with Gaussian shape depicted in Fig. 7(b). The accurate values were calculated analytically.



(a)



(b)

Fig. 7. Numerical integration.

Table 1. Numerical integration: error estimation.

$a_1 = 0.8,\ a_2 = 0.05$			$a_1 = 0.7925,\ a_2 = 0.0225$		
N	possibility	error/%	N	possibility	error/%
5	0.99202	21.78	5	0.99219	10.45
10	0.89476	9.97	10	0.99208	10.44
20	0.81189	$3.35 \cdot 10^{-1}$	20	0.92845	3.36
50	0.81467	$6.10 \cdot 10^{-3}$	50	0.89838	$8.14 \cdot 10^{-5}$
100	0.81462	$3.55 \cdot 10^{-7}$	100	0.89830	$6.58 \cdot 10^{-6}$
200	0.81462	$3.55 \cdot 10^{-7}$	200	0.89830	$6.58 \cdot 10^{-6}$

4.1. Reasoning

Apart from the motivational considerations above, employing restricted density functions has a positive side effect: the computational effort reduces considerably. Instead of handling samples of arbitrarily shaped density functions—and, therefore, searching through a high-dimensional solution space—here the dimension of the solution space reduces (depending on the chosen shape) to 2–4, respectively.

Owing to the reduced complexity of the optimisation problem, simple optimisation methods appear to be more suitable to perform the optimisation process. Therefore, two additional optimisation strategies have been implemented: a numerical version of the Gradient Search (GS) (Hermann, 2001), and Simulated Annealing (SA) (Nissen, 1997). The GS is performed as follows: starting from the current position in the solution space, a number of steps of certain length in different directions are undertaken si-

multaneously. The new potential solutions thus obtained are evaluated, and the best one is chosen as a new origin. If no further improvement is possible, the step-length is reduced. The algorithm terminates if no further improvement is achieved by employing a minimum step-length defined *a priori*. The modus operandi of SA differs mainly with respect to two aspects. Firstly, only one potential solution is computed each time. The direction of the corresponding step is determined at random. Secondly, small degradations of the solution's quality are accepted with a probability depending on the degree of degradation. The threshold for accepting degradations is scaled down with progressing computing time. Therefore, in later stages of the algorithm's execution, only improvements of solutions are possible, too. The main advantage of SA (in contrast to the GS) is a certain ability to escape from local optima.

4.2. Comparing Different Approaches

In Fig. 8 two different membership functions for h_{ave}—evaluated considering the Swedes example—are depicted. Thereby, $\mu_{h_{\mathrm{ave}1}}(\nu)$ was evaluated utilising the GA approach described in Sections 2 and 3, whereas $\mu_{h_{\mathrm{ave}2}}(\nu)$ was determined employing a restricted density function with Gaussian shape.

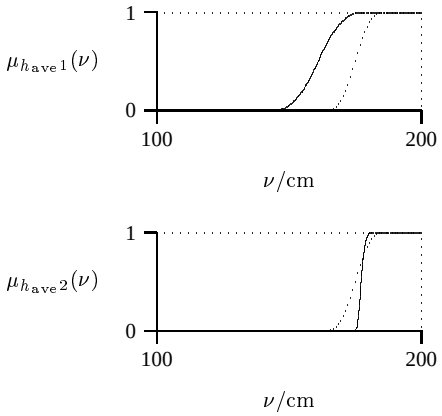


Fig. 8. Resulting membership functions for h_{ave} (in both the cases, μ_{tall} is marked with dotted line).

As was to be expected, the membership functions differ considerably. In the first case, where arbitrary density functions are allowed, the membership function reveals positive values for rather small average heights. This does not reflect our daily experience, and, especially, will not be confirmed by any real population in the world. In contrast, the restrictive approach yields a membership function which is in better conformance with human perceptions.

In the case of Version 2 of the Robert example, the situation is different. As has been mentioned before, $g_0(u)$ denotes the probability density function indicating the distribution of T_0, the time when Robert leaves his office. Now, a distribution like the one depicted in Fig. 9 may be absolutely possible—not only with regard to the generalised constraint induced by the corresponding proposition, but especially in reality, too. It is easily imaginable that Robert usually leaves his office at about 5:30 p.m., but sometimes about two hours later. Thus, it would not be useful to impose any restriction on the shape of g_0.

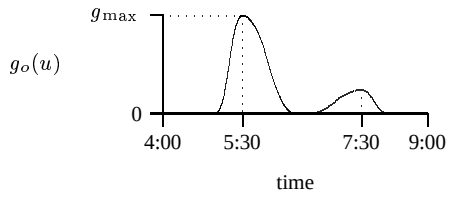


Fig. 9. Possible probability density function $g_0(u)$.

The question of whether or not the distribution of travel times g_1 should be restricted is not so easy to answer. If one is interested in a general answer to the given query, arbitrary shapes should, of course, be allowed. For instance, sporadic events like automobile accidents or route diversions may result in small isolated peaks. But possibly one is interested in an answer disregarding such exceptional events. In this case, employing a restricted density function could yield better results.

The crucial point is that a user of the methodological framework of CW has to decide in advance whether or not imposing a restriction on the shapes of the utilised density distributions would be recommendable. This decision may depend on various factors and, therefore, may not be easy to make. But it can, however, considerably influence the usability of the obtained results.

5. Conclusions

A Genetic Algorithm-based approach to solve the typical maximisation problem which arises when applying the generalised extension principle is described. Special attention is paid to transforming the given problem into a multi-objective form to provide universal applicability.

Based on the results obtained applying this GA, a motivation is given for employing restricted density functions (instead of arbitrarily shaped ones) when solving the maximisation problems considered. Furthermore, an application which provides different types of shapes for density functions as well as different algorithms for optimisation is described.

The question whether or not such restricted functions should be employed in a certain situation cannot be answered in general. But this very answer should be subjected to a careful examination.

References

Baker J. E. (1987): *Reducing bias and inefficiency in the selection algorithm*, In: Genetic Algorithms and their Applications: Proceedings of the 2nd International Conference on Genetic Algorithms (J. J. Grefenstette, Ed.). — Hillsdale, N.J.: Lawrence Erlbaum Associates, Inc., Publishers.

Gemeinder M. (2000): *Fuzzy constraint propagation with evolutionary algorithms*. — Proc. East West Fuzzy Colloquium, HS Zittau/Görlitz, Germany, pp. 196–203.

Gemeinder M. (2001a): *Restricted density functions for approximate reasoning on distributed quantities*. — Proc. 9th Zittau Fuzzy Colloquium, HS Zittau/Görlitz, Germany, pp. 29–35.

Gemeinder M. (2001B): *Computing with words: Multi-objective GAs for approximate reasoning*, In: Computational Intelligence, Theory and Applications, Proc. 7th Fuzzy Days, LNCS 2206 (B. Reusch, Ed.) — Berlin: Springer Verlag.

Hermann M. (2001): *Numerische Mathematik.* — München: Oldenbourg Wissenschaftsverlag.

Nissen V. (1997): *Einführung in evolutionäre Algorithmen: Optimierung nach dem Vorbild der Evolution.* — Braunschweig/Wiesbaden: Vieweg.

Pohlheim H. (1999): *Evolutionäre Algorithmen: Verfahren, Operatoren und Hinweise für die Praxis.* — Berlin: Springer Verlag.

Pozrikidis C. (1998): *Numerical Computation in Science and Engineering.* — Oxford: Oxford University Press, Inc.

Zadeh L. A. (1979): *A theory of approximate reasoning*. — Machine Intell., Vol. 9, pp. 149–194.

Zadeh L. A. (1999): *From computing with numbers to computing with words — From manipulation of measurements to manipulation of perceptions.* — IEEE Trans. Circ. Syst., Vol. 45, No. 1, pp. 105–119.

Zadeh L. A. (2001a): *The Robert Example.* — BISC Mailinglist, March 2001.

Zadeh L. A. (2001b): *A new direction in AI — Towards a computational theory of perceptions.* — 7th Fuzzy Days in Dortmund, Germany, Invited Lecture.