ORDERED WEIGHTED AVERAGING APPROACHES FOR AGGREGATING GRADUAL TRUST AND DISTRUST

Chris Cornelis[1] **Patricia Victor**[1] **Enrique Herrera-Viedma**[2]

[1] Dept. of Appl. Math. and Comp. Sci., Ghent University, Belgium, {Chris.Cornelis,Patricia.Victor}@UGent.be
[2] Dept. of Comp. Sci. and AI, University of Granada, Spain, viedma@decsai.ugr.es

Abstract

In this paper, we focus on the aggregation problem for (trust, distrust) couples in trust networks. In particular, we study approaches based on classical and induced ordered weighted averaging (OWA) operators.

Keywords: aggregation operators, trust networks, recommender systems, ordered weighted average

1 INTRODUCTION

In a (virtual) trust network, agents (nodes) can express their opinion about other agents through trust and distrust statements. In this paper, opinions are represented by means of trust scores, i.e., (trust, distrust) couples drawn from a bilattice [2]. When an agent a needs to establish an opinion about another, unknown agent x, it can inquire about x with one of its own trust relations, who in turn might consult a trust connection, etc., until an agent connected to x is reached. The process of predicting the trust score along the thus constructed path from a to x is called *trust propagation*. Since it often happens that a has not one, but several trust connections that it can consult for an opinion on x, we also require a mechanism for combining several trust scores originating from different sources. This process is called *trust aggregation* and it is very important to generate recommendations in recommender systems [3].

The problem of trust aggregation was first addressed in [1], in which we introduced a number of desirable criteria that a trust score aggregation operator should satisfy. We also presented a possible aggregation approach based on the use of Yager's Ordered Weighted Averaging (OWA, [4]) operators. In this paper, we undertake a more general study of the use of OWA operators for aggregating trust scores; in particular, we consider approaches based on classical OWA operators, as well as on induced ones [5].

The remainder of this paper is structured as follows: in Section 2, we introduce the necessary preliminaries about OWA operators and the trust score space we use, while in Section 3, we focus on the trust aggregation problem: we first recall a number of properties for trust score operators that were introduced in [1] (Section 3.1), and then in Section 3.2 set out to define a number of procedures based on standard OWA approaches and induced ones. Along the way, it will turn out that some properties need to be sacrificed, or at least adjusted. Finally, we offer a brief conclusion and point out further work in Section 4.

2 PRELIMINARIES

2.1 ORDERED WEIGHTED AVERAGING OPERATORS

The traditional OWA operator [4] models an aggregation process in which a sequence A of n scalar values are ordered decreasingly and then weighted according to their ordered position by means of a weighting vector $W = \langle w_i \rangle$, such that $w_i \in [0,1]$ and $\Sigma_i^n w_i = 1$. In particular, if c_i represents the i^{th} largest value in A,

$$OWA_W(A) = \sum_{i=1}^{n} w_i c_i \quad (1)$$

The OWA's main strength is its flexibility, since it enables us to model a whole range of aggregation strategies. Moreover, the reordering of the arguments introduces an element of non-linearity into an otherwise linear process.

With the induced OWA operator (IOWA, [5]), the ordering of the arguments is not based on their value, but on that of an induced ordering variable which is associated to them. In particular, let A and W be defined as above, and let I be a sequence of (not necessarily scalar) values, drawn from a linearly ordered space $(\mathcal{L}, \leq_{\mathcal{L}})$. If c_i represents the value in A associated with the i^{th} largest value in I,

$$IOWA_W(A, I, \leq_{\mathcal{L}}) = \sum_{i=1}^{n} w_i c_i \qquad (2)$$

Since the values of the induced ordering variable need not be scalars, IOWA operators offer even more flexibility than their standard counterparts.

2.2 TRUST AND DISTRUST: BILATTICE-BASED MODEL

In this paper, a trust network is represented as a directed graph (A, E, R) in which A is the set of agents (nodes), E is the set of trust connections (edges), and R is an $E \rightarrow [0,1]^2$ mapping that associates to each couple (a, b) of connected agents in E a trust score $R(a, b) = (R^+(a, b), R^-(a, b))$ in $[0,1]^2$, in which $R^+(a, b)$ is called the trust degree of a in b and $R^-(a, b)$ is called the distrust degree of a in b.

The set of trust scores can be endowed with a bilattice structure. In particular, the trust score space [2]

$$\mathcal{BL}^{\square} = ([0,1]^2, \leq_t, \leq_k, \neg)$$

consists of the set $[0,1]^2$ of trust scores, a trust ordering $\leq_t$, a knowledge ordering $\leq_k$, and a negation $\neg$ defined by

$$\begin{array}{rcl} (t_1, d_1) \leq_t (t_2, d_2) & \text{iff} & t_1 \leq t_2 \text{ and } d_1 \geq d_2 \\ (t_1, d_1) \leq_k (t_2, d_2) & \text{iff} & t_1 \leq t_2 \text{ and } d_1 \leq d_2 \\ \neg(t_1, d_1) & = & (d_1, t_1) \end{array}$$

for all (t_1, d_1) and (t_2, d_2) in $[0,1]^2$.

The "trust lattice" $([0,1]^2, \leq_t)$ orders the trust scores going from complete distrust $(0,1)$ to complete trust $(1,0)$. The "knowledge" lattice $([0,1]^2, \leq_k)$ evaluates the amount of available trust evidence, ranging from a "shortage of evidence", $t+d < 1$ (incomplete information), to an "excess of evidence", viz. $t+d > 1$ (inconsistent information). The boundary values of the $\leq_k$ ordering, $(0,0)$ and $(1,1)$, reflect ignorance, resp. contradiction. In this paper, trust scores (t, d) for which $t + d = 1$ are said to have perfect knowledge, while all others are called knowledge defective.

3 TRUST AGGREGATION

3.1 PROPERTIES

In [1], a number of desirable properties were introduced for a trust score aggregation operator $\Omega : ([0,1]^2)^n \rightarrow [0,1]^2$ $(n \geq 1)$, which are recalled below:

1. *Idempotence property* (**RQ1**).
$$\Omega((t, d), \ldots, (t, d)) = (t, d)$$
2. *Monotonicity property* (**RQ2**). Ω is monotonously increasing w.r.t. both $\leq_t$ and $\leq_k$, i.e., if $(t_j, d_j) \leq_t (t_j^{'}, d_j^{'})$, then $(p, q) \leq_t (p^{'}, q^{'})$; and if $(t_j, d_j) \leq_k (t_j^{'}, d_j^{'})$, then $(p, q) \leq_k (p^{'}, q^{'})$, with
$$\begin{array}{l} \Omega((t_1, d_1), \ldots, (t_j, d_j), \ldots, (t_n, d_n)) = (p, q) \\ \Omega((t_1, d_1), \ldots, (t_j^{'}, d_j^{'}), \ldots, (t_n, d_n)) = (p^{'}, q^{'}) \end{array}$$
3. *Commutativity property* (**RQ3**).
$$\begin{array}{c} \Omega((t_1, d_1), \ldots, (t_n, d_n)) = \\ \Omega((t_{\pi_1}, d_{\pi_1}), \ldots, (t_{\pi_n}, d_{\pi_n})) \end{array}$$
where π is any permutation of $\{1, \ldots, n\}$.
4. *Neutral element property* (**RQ4**). $(0,0)$ is neutral for Ω, i.e.,
$$\begin{array}{c} \Omega((t_1, d_1), \ldots, (t_{n-1}, d_{n-1}), (0,0)) = \\ \Omega((t_1, d_1), \ldots, (t_{n-1}, d_{n-1})) \end{array}$$
5. *Opposite arguments property* (**RQ5**). An equal number of $(1,0)$ and $(0,1)$ arguments yields contradiction, i.e.,
$$\Omega(\underbrace{(1,0), \ldots, (1,0)}_{n \text{ times}}, \underbrace{(0,1), \ldots, (0,1)}_{n \text{ times}}) = (1,1)$$

3.2 AGGREGATION OPERATORS

The application of an (induced) OWA operator requires scalar values as arguments. As such, OWA operators are not directly applicable to aggregate trust scores. Therefore, we propose to perform trust aggregation by means of two separate OWA operators, one for trust and one for distrust.

3.2.1 Standard OWA approaches

Below, we describe a generic procedure for applying (standard) OWA operators to the trust score aggregation problem:

1. Determine n, the number of trust score arguments distinct from $(0,0)$. Trust scores that represent complete ignorance do not take part in the aggregation process.[1]
2. Construct the sequences T and D, containing the n trust values (resp., the n distrust values) of the trust score arguments.

[1] If all trust scores equal $(0,0)$, the final result is also set to $(0,0)$ and the aggregation process terminates at this step.

3. Construct n-dimensional weight vectors W_T and W_D; weights may or may not be dependent on the actual trust score arguments.
4. Compute the aggregated trust score as $(OWA_{W_T}(T), OWA_{W_D}(D))$.

As it is clear from the above, the actual way of aggregating the trust scores is determined by the choice of the weight vectors. One strategy is to construct W_T and W_D beforehand. For instance, the final trust (resp., distrust) value can be evaluated as the extent to which a predefined fraction (at least one, all of them, a majority, ...) of the trust score arguments exhibits trust (resp., distrust).

Example 1 (Fixed weights) *In [1], given n trust score arguments to aggregate (all distinct from $(0,0)$), trust and distrust weights are computed by ($i = 1,\ldots,n$)*

$$W_{T_i} = \frac{2.\max(0, \lceil \frac{n}{2} \rceil - i + 1)}{\lceil \frac{n}{2} \rceil (\lceil \frac{n}{2} \rceil + 1)} \quad (3)$$

$$W_{D_i} = \frac{2.\max(0, \lceil \frac{n}{4} \rceil - i + 1)}{\lceil \frac{n}{4} \rceil (\lceil \frac{n}{4} \rceil + 1)} \quad (4)$$

The disparity between trust and distrust weights was motivated by the observation that a few distrust statements about x (in particular, a quarter of them) may suffice to reach a final conclusion of distrust, while the evaluation of trust depends on the majority of the arguments. Note that weights are decreasing, in a sense that the higher trust/distrust values have a stronger impact than the lower ones.

It can be verified, by construction and by the properties of an OWA operator, that **(RQ1)**, **(RQ3)** and **(RQ4)** always hold for this type of aggregation (in fact, regardless of the fact whether weights are fixed or not). In order for **(RQ5)** to hold, it suffices that $W_{T_i} = 0$ and $W_{D_i} = 0$ as soon as $i > \frac{n}{2}$. The proposal in Ex. 1 fulfills this condition. **(RQ2)** is a harder condition to fulfill in general, as the following example shows.

Example 2 *If the trust scores to aggregate are $(1,0)$, $(0,0)$, $(0,0)$ and $(0,0)$, then the outcome by our OWA procedure is $(1,0)$ (regardless of the choice of weights vectors, because $n = 1$). If we change these trust scores to $(1,0)$, $(0.1,0)$, $(0.1,0)$ and $(0.1,0)$, the number of arguments that take part in the OWA aggregation equals 4; computing the weights as in Ex. 1, i.e., $W_T = (\frac{2}{3}, \frac{1}{3}, 0, 0)$ and $W_D = (1,0,0,0)$, the final result of the aggregation equals $(0.7,0)$. So, although $(0,0) \leq_t (0.1,0)$ and $(0,0) \leq_k (0.1,0)$, $(1,0) \not\leq_t (0.7,0)$ and $(1,0) \not\leq_k (0.7,0)$.*

The reason for the failure of **(RQ2)** in this example is due to the presence (and subsequent alteration) of $(0,0)$ trust score arguments, which causes the application of the OWA operators to a different number of arguments. It can be verified, however, that if we add the restriction to **(RQ2)** that $(t_j, d_j) \neq (0,0)$ and $(t'_j, d'_j) \neq (0,0)$, the property holds, regardless of the weight vectors W_T and W_D, *provided they remain fixed.*

According to this analysis, fixed-weight OWA approaches perform well w.r.t. the criteria set out in Section 3.1. However, they also exhibit certain drawbacks, as the following example illustrates.

Example 3 *Assume the trust scores to aggregate are $(1,0)$, $(\delta,0)$, $(\delta,0)$ and $(\delta,0)$, with δ a value close to 0. In other words, three of the trust score arguments are very close to ignorance. Intuitively, one would expect their contribution to the final result to be very small. However, using the same weighting vector as in Ex. 1, the aggregated value will be $(\frac{2}{3} + \frac{1}{3}\delta, 0) \approx (\frac{2}{3}, 0)$, which differs significantly from $(1,0)$, the result obtained if the $(\delta,0)$ values are replaced by $(0,0)$.*

In fact, the above kind of problem occurs with any fixed-weight approach[2]; it is due to the fact that in this approach, trust scores are not discriminated w.r.t. the amount of knowledge they contain. Intuitively, one can argue that the closer a trust score (t,d) is to ignorance w.r.t. the $\leq_k$ order (i.e., $t + d \approx 0$), the lower its associated weight should be. On the other hand, it also makes sense to penalize trust scores (t,d) which are highly inconsistent (i.e., $t + d \approx 2$), to avoid that such "defective" trust scores influence the aggregation result in a disproportionate way (since both their trust and distrust values are high). In general, the "knowledge defect" of a trust score (t,d) can be expressed by the following formula:

$$kd(t,d) = 1 - |t + d - 1| \quad (5)$$

The following example illustrates a possible way to alter the weights based on the "knowledge defect" exhibited by the individual trust score arguments.

Example 4 (Knowledge-dependent weights) *Given n trust score arguments (t_i, d_i) ($i = 1,\ldots,n$, all trust scores distinct from (0,0)), we can associate with them a weight vector W^{kd} that represents each trust score's degree of knowledge defect relative to the*

[2]The only exception is when $W_{t_1} = W_{d_1} = 1$, i.e., only the highest trust and distrust values are taken into account.

remaining trust scores[3]*:*

$$W_i^{kd} = \frac{kd(t_i, d_i)}{\sum_{j=1}^{n} kd(t_j, d_j)} \qquad (6)$$

We cannot use W^{kd} directly as a weight vector inside the OWA operators, since the knowledge defect weights are not associated to ordered positions, but rather to the arguments themselves. We can however use them to modify existing OWA weight vectors W^T and W^D (which can be chosen as in the fixed weight approach). The final OWA weight vectors W_T and W_D are obtained as ($i = 1, \ldots, n$)

$$W_{T_i} = \frac{W_i^T W_{\pi_i}^{kd}}{\sum_{j=1}^{n} W_j^T W_{\pi_j}^{kd}}, W_{D_i} = \frac{W_i^D W_{\pi'_i}^{kd}}{\sum_{j=1}^{n} W_j^D W_{\pi'_j}^{kd}} \qquad (7)$$

in which π, π' represent the permutations that map an ordered position i to the index of the trust score that appears at that position.

To illustrate the operation of this approach, we apply it to the data of Ex. 3. In this case,

$$W^{kd} = \left(\frac{1}{1+3\delta}, \frac{\delta}{1+3\delta}, \frac{\delta}{1+3\delta}, \frac{\delta}{1+3\delta}\right) \qquad (8)$$

$$W_T = \left(\frac{2}{2+\delta}, \frac{\delta}{2+\delta}, 0, 0\right) \qquad (9)$$

The final aggregation result will be $\left(\frac{2+\delta^2}{2+\delta}, 0\right) \approx (1, 0)$, which corresponds to our intuition.

The application of knowledge-dependent OWA weights does not affect the properties **(RQ1)** and **(RQ3)**–**(RQ5)**, but **(RQ2)** cannot be maintained, not even in the weakened version which holds for fixed weights.

Example 5 *Consider the following trust score sequences:*

$$A = \langle (1,0), (0.9, 0.2), (0, 1) \rangle$$
$$B = \langle (1,0), (0.9, 0.1), (0, 1) \rangle$$
$$C = \langle (1,0.9), (0.9, 0.2), (0, 1) \rangle$$

Constructing the weight vectors as in Ex. 4, we obtain the aggregated trust scores $\left(\frac{281}{290}, 1\right)$ for A, $\left(\frac{29}{30}, 1\right)$ for B and $\left(\frac{101}{110}, 1\right)$ for C. However, while $(0.9, 0.2) \leq_t (0.9, 0.1)$, $\left(\frac{281}{290}, 1\right) \not\leq_t \left(\frac{29}{30}, 1\right)$ (comparing sequence A with B). Similarly, while $(1, 0) \leq_k (1, 0.9)$, $\left(\frac{281}{290}, 1\right) \not\leq_k \left(\frac{101}{110}, 1\right)$ (comparing sequence A with C).

[3]We assume that not all trust scores are equal to $(1, 1)$.

In this case, the failure of **(RQ2)** is due to the change of the knowledge-dependent weight vector W^{kd}. While it might be perceived as a disadvantage of this particular approach, it can also be argued that any attempt to penalize trust scores for their knowledge defects is incompatible with maintaining monotonicity. In fact, this is already evident in the fixed-weight approach: to guarantee that $(0, 0)$ can play its role as the neutral element of Ω, it needs to be handled separately.

A weakened version of trust monotonicity[4], however, can be obtained as follows:

Weak trust monotonicity property (**RQ2'**). If $(t_j, d_j) \leq_t (t_j^{'}, d_j^{'})$ and $t_j + d_j = t_j^{'} + d_j^{'}$, then $(p, q) \leq_t (p^{'}, q^{'})$, with

$$\Omega((t_1, d_1), \ldots, (t_j, d_j), \ldots, (t_n, d_n)) = (p, q)$$
$$\Omega((t_1, d_1), \ldots, (t_j^{'}, d_j^{'}), \ldots, (t_n, d_n)) = (p^{'}, q^{'})$$

As a corollary of this property, trust monotonicity holds in particular when all the involved trust scores (t, d) have perfect knowledge ($t + d = 1$).

3.2.2 Induced OWA approaches

In this section, we consider an alternative approach, based on IOWA operators, to take into account the knowledge defect exhibited by trust score arguments. As stated before, in order to use the IOWA approach, we require an order inducing variable that takes values drawn from a linearly ordered space. In this section, we will consider the trust scores themselves for this purpose. Since the bilattice orderings $\leq_t$ and $\leq_k$ are only partial rather than linear, they do not qualify to construct the required linearly ordered space. However, meaningful linear orderings over trust scores do exist. In particular, if we define, for (t_1, d_1), (t_2, d_2) in $[0, 1]^2$,

$$(t_1, d_1) \leq_{kd}^{t} (t_2, d_2) \text{ iff } (kd(t_1, d_1) < kd(t_2, d_2)) \vee (kd(t_1, d_1) = kd(t_2, d_2) \wedge t_1 \leq t_2) \qquad (10)$$

$$(t_1, d_1) \leq_{kd}^{d} (t_2, d_2) \text{ iff } (kd(t_1, d_1) < kd(t_2, d_2)) \vee (kd(t_1, d_1) = kd(t_2, d_2) \wedge d_1 \leq d_2) \qquad (11)$$

it can be verified that $([0, 1]^2, \leq_{kd}^{t})$ and $([0, 1]^2, \leq_{kd}^{d})$ are linearly ordered spaces. For both of them, the smallest element is $(0, 0)$, while $(1, 0)$ is the largest element for $\leq_{kd}^{t}$ and $(0, 1)$ the largest for $\leq_{kd}^{d}$.

The corresponding IOWA aggregation procedure is largely analogous to that for standard OWA:

[4]Note that a similar property for knowledge monotonicity holds only in the trivial case where $(t_j, d_j) = (t_j^{'}, d_j^{'})$, due to the condition $t_j + d_j = t_j^{'} + d_j^{'}$.

1. Determine n, the number of trust score arguments distinct from $(0, 0)$. Trust scores that represent complete ignorance do not take part in the aggregation process.[5]
2. Construct the sequences I, T and D; I contains the n trust scores, while T and D contain the corresponding n trust values (resp., the n distrust values).
3. Construct n-dimensional weight vectors W_T and W_D; weights may or may not be dependent on the actual trust score arguments.
4. Compute the aggregated trust score as $(IOWA_{W_T}(T, I, \leq^t_{kd}), IOWA_{W_D}(D, I, \leq^d_{kd}))$.

Note that while the IOWA approach allows to take into account knowledge defects by using the trust score orders $\leq^t_{kd}$ and $\leq^d_{kd}$, it still makes sense to use knowledge-dependent weights. This becomes evident when we apply the approach to the data in Ex. 3, which gives the same outcome as in the standard OWA case. On the other hand,the monotonicity property **(RQ2)** is not guaranteed for the IOWA approach, even if fixed weights are used. The following example illustrates this.

Example 6 *Consider the following trust score sequences:*

$$A = \langle(0.95, 0.05), (0.8, 0.2), (0, 0.5)\rangle$$
$$B = \langle(1, 0.05), (0.8, 0.2), (0, 0.5)\rangle$$

For sequence A it holds that $(0.95, 0.05) \geq^t_{kd} (0.8, 0.2) \geq^t_{kd} (0, 0.5)$ *and* $(0.8, 0.2) \geq^d_{kd} (0.95, 0.05) \geq^d_{kd} (0, 0.5)$, *while for sequence B,* $(0.8, 0.2) \geq^t_{kd} (1, 0.05) \geq^t_{kd} (0, 0.5)$ *and* $(0.8, 0.2) \geq^d_{kd} (1, 0.05) \geq^d_{kd} (0, 0.5)$. *Constructing the weight vectors as in Ex. 1, we obtain the aggregated trust scores* $(0.9, 0.8)$ *for A and* $\left(\frac{13}{15}, 0.8\right)$ *for B. However, while* $(0.95, 0.05) \leq_t (1, 0.05)$ *and* $(0.95, 0.05) \leq_k (1, 0.05)$, $(0.9, 0.8) \not\leq_t \left(\frac{13}{15}, 0.8\right)$ *and* $(0.9, 0.8) \not\leq_k \left(\frac{13}{15}, 0.8\right)$.

The IOWA approach does satisfy the weak trust monotonicity property **(RQ2')**, and this regardless of whether fixed or variable weight vectors are used.

4 CONCLUSION AND FUTURE WORK

In this paper, we have shown how ordered weighted averaging operators (OWA) can be used to solve the trust aggregation problem. In particular, we studied approaches where aggregation of trust scores is performed by applying a pair of standard or induced OWA operators to their trust and distrust components of the trust score separately.

We have also introduced two different ways to take into account that some trust scores are more useful than others in deriving a final aggregated trust score; specifically, the knowledge defect of a trust score can be incorporated both into the weight vector, as well as into the induced order used in an IOWA approach.

We also observed that taking into account knowledge defects is incompatible with maintaining the monotonicity of the trust and knowledge orders of the trust score bilattice; instead, we proposed a weaker property of trust monotonicity for trust scores that have the same level of knowledge defect, which holds for all the introduced approaches.

The choice as to which approach (OWA or IOWA, weight generating strategy, ...) is most suitable also depends on the application at hand. In our research, we are focusing in particular on recommender systems, i.e., applications which suggest items to users who might be interested in them. Trust information can help to establish more, and more accurate, recommendations, and its incorporation into existing recommender system technology is a topic of ongoing research (see e.g. [3]).

As another part of our future work, we also aim to refine the aggregation strategy to take into account certain aspects of the virtual trust network's topology. In particular, the current approach is indifferent as to the length of the paths that generated the individual trust scores, and also does not consider how many times the same agent appears on a path.

Acknowledgements

Chris Cornelis would like to thank the Research Foundation—Flanders for funding his research Patricia Victor would like to thank the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen) for funding her research. Enrique Herrera-Viedma would like to thank the financing of andalucian excellence project TIC05299, Feder Funds in FUZZYLING project (TIN2007-61079) and PETRI project (PET 2007-0460).

References

[1] P. Victor, C. Cornelis, M. De Cock, E. Herrera-Viedma. Aggregation of Gradual Trust and Distrust. *Proceedings of Eurofuse Workshop on Pref-*

[5] If all trust scores equal $(0, 0)$, the final result is also set to $(0, 0)$ and the aggregation process terminates at this step.

erence Modeling and Decision Analysis (Eurofuse 2009), Pág.259-264, 2009.

[2] P. Victor, C. Cornelis, M. De Cock, P. Pinheiro Da Silva. Gradual Trust and Distrust in Recommender Systems. *Fuzzy Sets and Systems* 160(10), Pág.1367-1382, 2009.

[3] P. Victor, C. Cornelis, M. De Cock, A. Teredesai. Trust- and Distrust-Based Recommendations for Controversial Reviews. *IEEE Intelligent Systems*, in press, 2009.

[4] R. R. Yager. On Ordered Weighted Averaging Aggregation Operators in Multicriteria Decision making. *IEEE Transactions on Systems, Man, and Cybernetics* 18, Pág.183-190, 1988.

[5] R. R. Yager, D. Filev. Induced Ordered Weighted Averaging Operators. *IEEE Transactions on Systems, Man, and Cybernetics* 29(2), Pág.141-150, 1999.