Computing Derivatives in Interval Type-2 Fuzzy Logic Systems

Jerry M. Mendel, Life Fellow, IEEE

Abstract—This paper makes type-2 fuzzy logic systems much more accessible to fuzzy logic system designers, because it provides mathematical formulas and computational flowcharts for computing the derivatives that are needed to implement steepest-descent parameter tuning algorithms for such systems. It explains why computing such derivatives is much more challenging than it is for a type-1 fuzzy logic system. It provides derivative calculations that are applicable to any kind of type-2 membership functions, since the calculations are performed without prespecifying the nature of those membership functions. Some calculations are then illustrated for specific type-2 membership functions.

Index Terms—Derivations, fuzzy logic system, type-2 fuzzy logic system.

I. INTRODUCTION

A type-2 fuzzy logic system (FLS) lets us directly model (and, subsequently, minimize the effects of) a variety of uncertainties1 that cannot be directly modeled using a type-1 FLS. The price paid for being able to do this is somewhat greater complexity for a type-2 FLS than for a type-1 FLS; but, if one is unable to achieve satisfactory performance—in the face of uncertainties—using a type-1 FLS, then this may be a small price to pay for the improved performance. Of course, to achieve the improved performance one must first be able to design a type-2 FLS. Although there are different approaches to doing this, the most popular one-to-date uses steepest descent algorithms (also referred to as back-propagation algorithms) for adjusting all design parameters, and such algorithms require the computation of first-derivatives of an objective function with respect to each and every design parameter. The purpose of this paper is to provide such first-derivative formulas since they do not appear in the existing literature. Doing this will make type-2 FLSs much more accessible to FLS designers. To begin, we review the essence of a type-2 FLS.

A type-2 FLS (just as a type-1 FLS) contains four components: rules, fuzzifier, inference engine and output-processor (Fig. 1). During the operation of a type-2 FLS, measurements activate the fuzzifier, inference engine and output processor blocks in that order. The output processor contains two components: type-reduction and defuzzification. When arbitrary type-2 fuzzy sets2 are used a type-2 FLS is computationally prohibitive. On the other hand, when all type-2 fuzzy sets are modeled as interval sets, then we obtain an interval type-2 FLS, and such FLSs are very practical.3

Depending upon the way in which input measurements (x) to the FLS are fuzzified, either as singletons, type-1 fuzzy numbers or type-2 fuzzy numbers, three kinds of interval type-2 FLSs are possible: interval singleton type-2 FLS, interval type-1 nonsingleton type-2 FLS, and interval type-2 nonsingleton type-2 FLS. In the main body of this paper, we focus on an interval singleton type-2 FLS, but provide the extensions of our results to an interval type-2 nonsingleton type-2 FLS (which contains an interval type-1 nonsingleton type-2 FLS as a special case) in Appendix B.

The inference engine first produces a firing set, which is then used to produce an output consequent set membership function (MF) for each fired rule, which can then be used to produce a MF for all (combined) fired rules. For an interval singleton type-2 FLS, it is possible to obtain closed-form formulas for all of these quantities, and these results are given in Section A.2 of Appendix A; they are obtained by using well-known closed-form formulas for the join and meet of interval sets (e.g., [11] and [6]).

The type-reducer leads to a type-reduced set that provides an interval of uncertainty for the output of a type-2 FLS that is analogous to a confidence interval that provides an interval of uncertainty for a probabilistic system. The more uncertainties that occur in a type-2 FLS, which translate into more uncertainties about its MFs, the larger will be the type-reduced set, and vice versa. Regardless of which type-reduction method4 we choose, the type-reduced set for an interval type-2 FLS is an interval type-1 set, and its two end-points can be computed using an exact iterative method developed by Karnik and Mendel [2] whose steps are listed in Section A.3 of Appendix A. Because the type-reduced set is an interval set, its defuzzified value is simply the average value of its two end-points. A formula for the defuzzified output is given in Section A.4 of Appendix A. Using the formulas given in Appendix A, it is possible to compute the input-output relation, y = f_{l}(x), of an interval singleton type-2 FLS, and these formulas are the starting point for the design of such a FLS.

It is well known that a type-1 FLS is characterized by a fuzzy basis function (FBF) expansion (e.g., [8]), and that such an expansion is not only useful for computing the output of that FLS but also used during its design, especially as the starting point

1Because these uncertainties have been enumerated many times before, we refer the reader to [6, p. 68] for a list of them, as well as for discussions about them.

2See Section A.1 of Appendix A for some important definitions about type-2 fuzzy sets.

3See [7] for many reasons supporting the use of interval type-2 sets.

4Type-reduction is briefly reviewed in Section A.3 of Appendix A.
for computing derivatives of an objective function with respect to MF parameters. An interval singleton type-2 FLS is characterized by two fuzzy basis function (FBF) expansions [5], one associated with the left end-point of the type-reduced set, and the other associated with the right end-point of the type-reduced set. Unlike the FBF expansion for a type-1 FLS, the FBF expansions for an interval singleton type-2 FLS cannot be used to actually compute the left and right end-points of the type-reduced set, because the latter are in terms of two crossover points, \(L(x)\) and \(R(x)\) (\(L\) and \(R\) for short), that are computed using the Karnik–Mendel iterative procedures. By the end of those procedures not only are \(L\) and \(R\) computed but so also are the left and right end-points of the type-reduced set. Interestingly enough, the formulas for the two FBF expansions can however be used during the design of the FLS, as we explain below.

By “design” we mean specify or optimize the parameters that characterize the interval type-2 FLS. A type-2 FLS design method is associated with the following design problem.

We are given a collection of \(N\) input-output numerical data training pairs, \(\{x^{(1)} : y^{(1)}\}, \{x^{(2)} : y^{(2)}\}, \ldots, \{x^{(N)} : y^{(N)}\}\), where \(x\) is the vector input and \(y\) is the scalar output of an interval singleton type-2 FLS. Our goal is to completely specify this type-2 FLS using the training data.

A design method establishes how to specify all the parameters of the antecedent and consequent membership functions using the training pairs \(\{x^{(1)} : y^{(1)}\}, \{x^{(2)} : y^{(2)}\}, \ldots, \{x^{(N)} : y^{(N)}\}\). The most popular design method—the back-propagation method—is one in which all MF parameters are tuned using a steepest descent algorithm whose general form is

\[
\theta(n+1) = \theta(n) - \alpha \frac{\partial \epsilon(n)}{\partial \theta} \bigg|_{\theta}
\]

where \(\theta\) denotes any one of the FLS design parameters

\[
\epsilon(n) = \frac{1}{2} \left[ f_{\theta} \left( x^{(t)} \right) - y^{(t)} \right]^2, \quad t = 1, \ldots, N
\]

and \([\theta]\) indicates that after taking the derivative of \(\epsilon(n)\) with respect to a specific \(\theta\) we must replace all remaining \(\theta\) values by \(\theta(n)\). The challenge to developing easy-to-use steepest descent algorithms is to establish formulas for the derivatives \(\partial \epsilon(n)/\partial \theta\).

Generally, it is much more complicated to compute such derivatives for an interval type-2 FLS than it is for a type-1 FLS, because of the following.

- In an interval singleton type-2 FLS the design parameters appear in upper and lower MFs, whereas in a singleton type-1 FLS they appear in a single MF.
- In an interval singleton type-2 FLS, type-reduction establishes the two parameters \(L(x)\) and \(R(x)\), which in turn establish the upper and lower firing-interval MFs that are used to compute the left and right end-points of the type-reduced set [see (A-16) and (A-17)]. There is no type-reduction in a type-1 FLS.

In the rest of this paper we establish mathematical formulas to compute the derivatives \(\partial \epsilon(n)/\partial \theta\). The derivations of these formulas can be approached in different ways, e.g. choose a type-2 fuzzy set’s membership function footprint of uncertainty \(^6\) (FOU) as soon as possible or defer the choice of a FOU for as long as possible. In this paper, we take the latter approach, because by doing so our results are applicable to any kind of FOU.

Section II provides some fundamental assumptions; Section III provides general formulas for the right and left end-points of the type-reduced set; Section IV provides formulas for derivatives of \(\epsilon(n)\) with respect to antecedent MF parameters; Section V provides formulas for derivatives of \(\epsilon(n)\) with respect to consequent MF parameters; Section VI provides an example; Section VII provides conclusions. Note that the formulas in Sections III–V are independent of the choices made for the type-2 antecedent and consequent MFs. Finally, Appendix A has important background material about type-2 fuzzy sets, and fuzzy inference engine, type-reduction and defuzzification for interval singleton type-2 FLSs; and, Appendix B presents derivative formulas for interval type-2 nonsingleton type-2 FLSs.

\(^3\)Upper and lower MFs are defined in Definition A-8.

\(^6\)The FOU is defined in Definition A-6.
IEEE TRANSACTIONS ON FUZZY SYSTEMS, VOL. 12, NO. 1, FEBRUARY 2004

TABLE I

RULE-ORDERED AND RULE-REORDERED QUANTITIES

<table>
<thead>
<tr>
<th>Calculations</th>
<th>Rule-Ordered</th>
<th>Rule-Reordered</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$y_i = (y'_1, \ldots, y'_M)^T$</td>
<td>$y'_i = (y'_1, \ldots, y'_M)^T$</td>
</tr>
<tr>
<td>Left-end calculations(^b)</td>
<td>$y_i = Q_i z_i$</td>
<td>$y_i = Q_i z_i$</td>
</tr>
<tr>
<td>$y_i$</td>
<td>$y_i = \sum_{j=1}^{L} \tilde{f}^j z_i^j + \sum_{j=L+1}^{M} \tilde{f}^j z_i^j$</td>
<td>$y'<em>i = \sum</em>{j=1}^{L} \tilde{g}^j y_i^j + \sum_{j=L+1}^{M} \tilde{g}^j y_i^j$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Right-end calculations(^b)</td>
<td>$y_r = Q_r z_r$</td>
<td>$y_r = Q_r z_r$</td>
</tr>
<tr>
<td>$y_r$</td>
<td>$y_r = \sum_{j=1}^{R} \tilde{f}^j z_r^j + \sum_{j=R+1}^{M} \tilde{f}^j z_r^j$</td>
<td>$y'<em>r = \sum</em>{j=1}^{R} \tilde{g}^j y_r^j + \sum_{j=R+1}^{M} \tilde{g}^j y_r^j$</td>
</tr>
</tbody>
</table>

a. There are many kinds of permutation matrices. The kind we use has columns that are elementary vectors (i.e., vectors all of whose elements are zero except one element that is one) and these vectors are arranged—permuted—so as to move elements in the original matrix or vector to new locations in the transformed matrix or vector.

b. Note that in [2] and [5], $f' (x)$ and $f' (x)$ are used to denote both the rule-ordered and rule reordered lower and upper firing intervals. This does not cause any problems, unless—as in the present paper—formulas for $y_i$ and $y_r$ are used for further computations.

II. ASSUMPTIONS

We make four fundamental assumptions.

1) Parameters to be tuned are different for each rule and for each antecedent and consequent, i.e. no parameters are shared across rules or MFs.

2) Formulas for antecedent and consequent MFs are not specified ahead of time.

3) Derivatives needed for a steepest descent tuning-algorithm are to be computed by means of mathematical formulas.

4) Center-of-sets type-reduction is used.

If some parameters are shared across rules or MFs then at some point in our analyzes below a detour must be taken. We will indicate precisely where this occurs. By not specifying formulas for antecedent and consequent MFs ahead of time, our results will be as general as possible. If mathematical formulas for derivatives cannot be obtained, it may still be possible to determine derivatives numerically using perturbation techniques.

We do not cover such techniques in this paper because the kinds of primary MFs that one usually chooses can be described mathematically, e.g., triangles, trapezoids, Gaussians, etc. We choose center-of-sets type-reduction because there is an explicit appearance of antecedent and consequent MF parameters for it. The same is true for height type-reduction but is not true for centroid or center-of-sums type-reduction (see Table A-I).

III. GENERAL EXPRESSIONS FOR $y_i$ AND $y_r$

Although, as discussed in Section I, we always compute $y_i$ and $y_r$ using the Karnik–Mendel iterative procedures, we use formulas for $y_i$ and $y_r$ to compute derivatives that are needed in the back-propagation update algorithms. Such formulas can be derived from step 4 of the iterative procedure (Section A.3), the paragraph just below that procedure, and (A-16) and (A-17), and are

$$y_i = \sum_{j=1}^{L} \tilde{g}^j y_i^j + \sum_{j=L+1}^{M} \tilde{g}^j y_i^j$$

and

$$y_r = \sum_{j=1}^{R} \tilde{g}^j y_r^j + \sum_{j=R+1}^{M} \tilde{g}^j y_r^j$$

These formulas cannot be used as is because the $f'_r$, $y'_i$, $f'_r$, and $y'_i$ have been reordered during step-1 of the two iterative procedures used to compute $y_i$ and $y_r$. In order to compute derivatives of $y_i$ and $y_r$ with respect to MF parameters, we need to know exactly where specific antecedent and consequent MF parameters are located, and this is very difficult to ascertain when $y_i$ and $y_r$ are not in rule-ordered format. So, our first task is to re-express
TABLE II
FORMULAS USED TO CALCULATE \( y_h \) AND \( y_r \) IN RULE-ORDERED FORMAT

<table>
<thead>
<tr>
<th>( y_t ) calculations</th>
<th>( y_r ) calculations</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f = (f^1, f^2, ..., f^M)^T )</td>
<td>( \tilde{f} = (\tilde{f}^1, \tilde{f}^2, ..., \tilde{f}^M)^T )</td>
</tr>
<tr>
<td>( g = (g^1, g^2, ..., g^N)^T = Q_s f )</td>
<td>( \tilde{g} = (\tilde{g}^1, \tilde{g}^2, ..., \tilde{g}^M)^T = Q_s \tilde{f} )</td>
</tr>
<tr>
<td>( \mathbf{E}<em>i = (e</em>{i1}</td>
<td>e_{i2}</td>
</tr>
<tr>
<td>( \mathbf{E}_r = (0</td>
<td>...</td>
</tr>
<tr>
<td>( \mathbf{M}_{11} = Q^T_s \mathbf{E}_1 \mathbf{Q}_s M \times M )</td>
<td>( \mathbf{M}_{11} = Q^T_s \mathbf{E}_1 \mathbf{Q}_s M \times M )</td>
</tr>
<tr>
<td>( \mathbf{M}_{12} = Q^T_s \mathbf{E}_2 \mathbf{Q}_s M \times M )</td>
<td>( \mathbf{M}_{12} = Q^T_s \mathbf{E}_2 \mathbf{Q}_s M \times M )</td>
</tr>
<tr>
<td>( \mathbf{M}_{21} = (1, 1, ..., 1, 0, 0, ..., 0)^T M \times 1 )</td>
<td>( \mathbf{r}_r = (1, 1, ..., 1, 0, 0, ..., 0)^T M \times 1 )</td>
</tr>
<tr>
<td>( \mathbf{M}_{22} = L \times M )</td>
<td>( \mathbf{s}_r = (0, 0, ..., 0, 1, ..., 1)^T M \times 1 )</td>
</tr>
<tr>
<td>( \mathbf{a}<em>i = \mathbf{M}</em>{1i} \mathbf{z}_i M \times 1 )</td>
<td>( \mathbf{a}<em>i = \mathbf{M}</em>{1i} \mathbf{z}_i M \times 1 )</td>
</tr>
<tr>
<td>( \mathbf{b}<em>i = \mathbf{M}</em>{2i} \mathbf{z}_i M \times 1 )</td>
<td>( \mathbf{b}<em>i = \mathbf{M}</em>{2i} \mathbf{z}_i M \times 1 )</td>
</tr>
<tr>
<td>( \mathbf{c}_r^T = \mathbf{r}_r^T \mathbf{Q}_s 1 \times M )</td>
<td>( \mathbf{c}_r^T = \mathbf{r}_r^T \mathbf{Q}_s 1 \times M )</td>
</tr>
<tr>
<td>( \mathbf{d}_r^T = \mathbf{s}_r^T \mathbf{Q}_s 1 \times M )</td>
<td>( \mathbf{d}_r^T = \mathbf{s}_r^T \mathbf{Q}_s 1 \times M )</td>
</tr>
</tbody>
</table>

a. Using the same symbols, \( g \) and \( \tilde{g} \), for both \( y_t \) and \( y_r \) calculations will not cause any problems in later calculations.

(3) and (4) in rule-ordered format. Along the way, we shall also remove the explicit dependence of \( y_h \) on \( L \) and \( y_r \) on \( R \).

**Rule-ordered firing intervals** are denoted as \( F^i(\mathbf{x}') \). We have labeled the rule reordered firing intervals \( G^i(\mathbf{x}') \), i.e.,

\[
F^i(\mathbf{x}') = [f^i, f^i']
\]

\[
G^i(\mathbf{x}') = [g^i, g^i']
\]

In (A-15), observe that \( q^j = [q^j_1, q^j_2](i = 1, ..., M) \), and that in the two Karnik–Mendel procedures it is the \( q^j_1 \) (\( i = 1, ..., M \)) that are reordered in order to compute \( y_h \), whereas it is the \( q^j_2 \) (\( i = 1, ..., M \)) that are reordered to compute \( y_r \). In this paper, we continue to let \( q^j_1 \) and \( q^j_2 \) denote *rule-reordered* values; however, we introduce \( z^j_1 \) and \( z^j_2 \) to denote their *rule-ordered* counterparts. Table I summarizes the rule-ordered and rule-unordered quantities that we will need in the rest of this paper. The question that we address next is how do we go from the rule-reordered versions of \( y_h \) and \( y_r \) to the rule-ordered versions?

**A. \( y_h \) Re-Expressed in Rule-Ordered Format**

We want to re-express \( y_h \), given in rule-reordered form (Table I), in terms of a rule-ordered quantities, i.e., in terms of \( f^i, f^i', z^j_1 \) and \( z^j_2 \). To begin, we define a collection of vectors and matrices that are summarized in Table II.

**Fact 1:** \( y_h \) can be re-expressed in terms of rule-ordered quantities as

\[
y_h = \frac{\mathbf{f}^T \mathbf{M}_{12} \mathbf{z}_4 + \mathbf{f}^T \mathbf{M}_{22} \mathbf{z}_4}{\mathbf{f}^T \mathbf{Q}_s \mathbf{f} + \mathbf{s}_r^T \mathbf{Q}_s \mathbf{f}}
\]

\[
= \frac{\mathbf{f}^T \mathbf{a}_i + \mathbf{f}^T \mathbf{d}_r}{\mathbf{f}^T \mathbf{c}_r + \mathbf{d}_r^T \mathbf{f}}
\]

\[
= \frac{\sum_{i=1}^{M} \alpha_{ih} f^i + \sum_{i=1}^{M} \beta_{ih} f^i}{\sum_{i=1}^{M} \alpha_{rh} f^i + \sum_{i=1}^{M} \beta_{rh} f^i}.
\]

**Proof:** We must re-express the four sums that appear in the rule-reordered version of \( y_h \), given in Table I, i.e.,

\[
\sum_{i=1}^{L} q^j_1 y^j_1 = (\mathbf{E}_1 \mathbf{Q}_s)^T (\mathbf{E}_1 \mathbf{Y}_1)
\]

\[
= (\mathbf{E}_1 \mathbf{Q}_s)^T (\mathbf{E}_1 \mathbf{Q}_s \mathbf{z}_4)
\]

\[
= \mathbf{f}^T \mathbf{M}_{12} \mathbf{z}_4
\]

\[
= \sum_{j=L+1}^{M} q^j_1 y^j_1 = (\mathbf{E}_2 \mathbf{Q}_s)^T (\mathbf{E}_2 \mathbf{Y}_1)
\]

\[
= (\mathbf{E}_2 \mathbf{Q}_s)^T (\mathbf{E}_2 \mathbf{Q}_s \mathbf{z}_4)
\]

\[
= \mathbf{f}^T \mathbf{M}_{22} \mathbf{z}_4.
\]

Note, for example, that \( (\mathbf{E}_1 \mathbf{Q}_s)^T = (\mathbf{E}_1 \mathbf{Q}_s)^T (\mathbf{E}_2 \mathbf{Q}_s \mathbf{z}_4) \) and \( \mathbf{E}_1 \mathbf{Y}_1 = (\mathbf{E}_1 \mathbf{Q}_s)^T (\mathbf{E}_1 \mathbf{Q}_s \mathbf{z}_4) \), so that \( \sum_{i=1}^{L} q^j_1 y^j_1 = (\mathbf{E}_1 \mathbf{Q}_s)^T (\mathbf{E}_2 \mathbf{Y}_1) \).
Substitute (8)–(11) into \((\sum_{i=1}^{L} g_i^T \bar{g} + \sum_{j=L+1}^{M} d_j^T \bar{d}_j)/(\sum_{i=1}^{L} g_i^T + \sum_{j=L+1}^{M} d_j^T)\) to obtain the first term on the right-hand side of (7). The second term appearing on the right-hand side of (7) is obtained by using the quantities \(\alpha_i, \beta_i, c_f^j,\) and \(d_f^j\), that are defined in the last four rows of Table II, in the first term on the right-hand side of (7). The last term on the right-hand side of (7) is just an expanded version of the second term.

Observe that (7) involves the entire \(\bar{f}\) and \(\bar{f}\) vectors and the entire \(\bar{z}\) vector. This is good because we can then take the derivatives of \(\bar{y}\) with respect to any element in \(\bar{f}\) or \(\bar{f}\) without having to worry ahead of time whether or not it actually appears in \(\bar{y}\).
TABLE III

<table>
<thead>
<tr>
<th>$x_k$</th>
<th>$\bar{\mu}_i(x_k)$</th>
<th>$\mu_j^l(x_k)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_k &lt; m_{i1}'$</td>
<td>$m_{i1}'$, $\sigma_i^k$</td>
<td>$m_{i2}'$, $\sigma_i^k$</td>
</tr>
<tr>
<td>$m_{i1}' \leq x_k \leq (m_{i1}' + m_{i2}')/2$</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>$(m_{i1}' + m_{i2}')/2 \leq x_k \leq m_{i2}'$</td>
<td>$m_{i1}'$, $\sigma_i^k$</td>
<td>$m_{i1}'$, $\sigma_i^k$</td>
</tr>
<tr>
<td>$x_k &gt; m_{i2}'$</td>
<td>$m_{i2}'$, $\sigma_i^k$</td>
<td>$m_{i2}'$, $\sigma_i^k$</td>
</tr>
</tbody>
</table>

Matrices $M_{i1}$ and $M_{i2}$ and vectors $r_i$ and $s_i$ will automatically dispose of the unnecessary elements of $\bar{f}$ and $\underline{f}$, since they depend on $L$.

B. $y_r$: Re-Expressed in Rule-Ordered Format

**Fact 2:** $y_r$ can be re-expressed in terms of rule-ordered quantities as

$$y_r = \bar{f}^T M_{i1} z_r + \bar{f}^T M_{i2} \bar{z}_r + \bar{f}^T Q \bar{z}_r + \bar{f}^T s_i^T Q \bar{f}$$

$$= \bar{f}^T M_{i1} z_r + \bar{f}^T M_{i2} \bar{z}_r$$

$$= \sum_{l=1}^M a_{r,l} f_l^T + \sum_{l=1}^M b_{r,l} f_l$$

$$= \sum_{l=1}^M c_r f_l^T + \sum_{l=1}^M d_{r,l} f_l,$$  \hspace{1cm} (12)

**Proof:** Just follow the proof of Fact 1 using quantities that are defined in the right-hand column of Table II.

**IV.** CALCULATION OF $\partial f_{k,m}^{i,j}/\partial \theta_{k,m}^{i,j}$ FOR ANTECEDENT PARAMETERS

Antecedent parameters are the parameters that characterize antecedent MFs. For example, a Gaussian primary MF with uncertain mean, as defined in (A-7), is characterized by three parameters, $m_1$, $m_2$, and $\sigma$. Temporarily, let us denote any one of the antecedent parameters that will be tuned as $\theta_{k,m}^{i,j}$ ($k = 1, \ldots, p$ and $l = 1, \ldots, M$). Index $m$ denotes the fact that there can be more than one parameter associated with the MF of each antecedent ($k$) and rule ($l$). Here we use the chain rule to compute $\partial f_{k,m}^{i,j}/\partial \theta_{k,m}^{i,j}$. Our starting point is (2), in which $f_{k,m}(x)$ is given by (A-18), which we restate here as

$$f_{k,m}(x) = \frac{\eta_r(x) + \eta_l(x)}{2}.$$  \hspace{1cm} (13)

Hence

$$\frac{\partial f_{k,m}^{i,j}}{\partial \theta_{k,m}^{i,j}} = \frac{\partial f_{k,m}^{i,j}}{\partial f_{k,m}^{i,j}} \left[ \frac{\partial f_{k,m}^{i,j}}{\partial \eta_r} + \frac{\partial f_{k,m}^{i,j}}{\partial \eta_l} \right]$$

$$= \frac{1}{2} \left[ f_{k,m}^{i,j}(x) - y^{i,j} \right] \left[ \frac{\partial \eta_r}{\partial \theta_{k,m}^{i,j}} + \frac{\partial \eta_l}{\partial \theta_{k,m}^{i,j}} \right]$$  \hspace{1cm} (14)

where we used the facts that $\partial f_{k,m}^{i,j}/\partial f_{k,m}^{i,j} = \left[ f_{k,m}^{i,j}(x) - y^{i,j} \right]$ and $\partial f_{k,m}(x)/\partial \eta_l = \partial f_{k,m}(x)/\partial \eta_l = 1/2$. In (14), we now treat $\eta_r$ and $\eta_l$ as functions of $\bar{f}$ and $\underline{f}$; hence

$$\frac{\partial \eta_l}{\partial \theta_{k,m}^{i,j}} = \sum_{l=1}^M \left( \frac{\partial \eta_l}{\partial \bar{f}^j} \frac{\partial \bar{f}^j}{\partial \theta_{k,m}^{i,j}} + \frac{\partial \eta_l}{\partial \underline{f}^j} \frac{\partial \underline{f}^j}{\partial \theta_{k,m}^{i,j}} \right)$$  \hspace{1cm} (15)

$$\frac{\partial \eta_r}{\partial \theta_{k,m}^{i,j}} = \sum_{l=1}^M \left( \frac{\partial \eta_r}{\partial \bar{f}^j} \frac{\partial \bar{f}^j}{\partial \theta_{k,m}^{i,j}} + \frac{\partial \eta_r}{\partial \underline{f}^j} \frac{\partial \underline{f}^j}{\partial \theta_{k,m}^{i,j}} \right).$$  \hspace{1cm} (16)

We now need to evaluate all of the derivatives in (15) and (16).

**Fact 3:** The following are true:

$$\frac{\partial \eta_l}{\partial \bar{f}^j} = a_{r,l} - y_{r,a_{r,l}}$$  \hspace{1cm} (17)

$$\frac{\partial \eta_l}{\partial \underline{f}^j} = \bar{f}^T c + \bar{f}^T d_i$$

$$= \frac{\partial \eta_l}{\partial \bar{f}^j} = \bar{f}^T c + \bar{f}^T d_i$$

$$\frac{\partial \eta_l}{\partial \theta_{k,m}^{i,j}} = b_{r,l} - y_{r,b_{r,l}}$$  \hspace{1cm} (18)

$$\frac{\partial \eta_r}{\partial \bar{f}^j} = \bar{f}^T c + \bar{f}^T d_i$$

$$\frac{\partial \eta_r}{\partial \underline{f}^j} = \bar{f}^T c + \bar{f}^T d_i$$

$$\frac{\partial \eta_r}{\partial \theta_{k,m}^{i,j}} = c_{r,l} - y_{r,c_{r,l}}$$  \hspace{1cm} (19)

$$\frac{\partial \eta_r}{\partial \theta_{k,m}^{i,j}} = \frac{\partial \eta_r}{\partial \bar{f}^j} = \bar{f}^T c + \bar{f}^T d_i$$

$$\frac{\partial \eta_r}{\partial \theta_{k,m}^{i,j}} = \frac{\partial \eta_r}{\partial \underline{f}^j} = \bar{f}^T c + \bar{f}^T d_i$$

$$= \frac{\partial \eta_r}{\partial \theta_{k,m}^{i,j}}$$  \hspace{1cm} (20)

**Proof:** Because all calculations are alike, we only provide the derivation of (18). From the second and third forms of $y_r$ in (7), it follows that

$$\frac{\partial \eta_l}{\partial \bar{f}^j} = \frac{(\bar{f}^T c + \bar{f}^T d_i) b_{r,l} - (\bar{f}^T c + \bar{f}^T d_i) b_{r,l}}{(\bar{f}^T c + \bar{f}^T d_i)^2}$$  \hspace{1cm} (21)

so that

$$\frac{\partial \eta_l}{\partial \theta_{k,m}^{i,j}} = \frac{b_{r,l} - y_{r,b_{r,l}}}{\bar{f}^T c + \bar{f}^T d_i} = \frac{b_{r,l} - y_{r,b_{r,l}}}{\bar{f}^T c + \bar{f}^T d_i}$$

Q.E.D.
TABLE IV
Derivatives of $p^i_k(x_k)$

<table>
<thead>
<tr>
<th>$x_k$</th>
<th>$\partial \tilde{\mu}^i_k(x_k) / \partial \theta^i_{k1}$</th>
<th>$\partial \tilde{\mu}^i_k(x_k) / \partial \theta^i_{k2}$</th>
<th>$\partial \tilde{\mu}^i_k(x_k) / \partial \theta^i_{k3}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_k &lt; m^i_{11}$</td>
<td>$(x - m^i_{11})N(m^i_{11}, \sigma; x)/\sigma^2$</td>
<td>0</td>
<td>$(x - m^i_{11})^2 N(m^i_{11}, \sigma; x)/\sigma^3$</td>
</tr>
<tr>
<td>$m^i_{11} \leq x_k \leq (m^i_{11} + m^i_{12})/2$</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$(m^i_{11} + m^i_{12})/2 \leq x_k \leq m^i_{12}$</td>
<td>0</td>
<td>$(x - m^i_{12})N(m^i_{12}, \sigma; x)/\sigma^2$</td>
<td>$(x - m^i_{12})^2 N(m^i_{12}, \sigma; x)/\sigma^3$</td>
</tr>
<tr>
<td>$x_k &gt; m^i_{12}$</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

TABLE V
Derivatives of $p^j_k(x_k)$

<table>
<thead>
<tr>
<th>$x_k$</th>
<th>$\partial \tilde{\mu}^j_k(x_k) / \partial \theta^j_{k1}$</th>
<th>$\partial \tilde{\mu}^j_k(x_k) / \partial \theta^j_{k2}$</th>
<th>$\partial \tilde{\mu}^j_k(x_k) / \partial \theta^j_{k3}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_k \leq (m^j_{11} + m^j_{12})/2$</td>
<td>0</td>
<td>$(x - m^j_{11})N(m^j_{11}, \sigma; x)/\sigma^3$</td>
<td>$(x - m^j_{11})^2 N(m^j_{11}, \sigma; x)/\sigma^3$</td>
</tr>
<tr>
<td>$x_k &gt; (m^j_{11} + m^j_{12})/2$</td>
<td>$(x - m^j_{11})N(m^j_{11}, \sigma; x)/\sigma^3$</td>
<td>0</td>
<td>$(x - m^j_{11})^2 N(m^j_{11}, \sigma; x)/\sigma^3$</td>
</tr>
</tbody>
</table>

Next, we focus on computing $\partial f^j / \partial \theta^j_{km}$ and $\partial f^j / \partial \alpha_{km}$ that are needed in (15) and (16). Formulas for $f^j(x)$ and $f^j(x)$ are given in (A-11) and (A-12), respectively. Observe that the former are in terms of the antecedent upper MFs $\tilde{p}^j_k$, whereas the latter are in terms of the antecedent lower MFs $\tilde{p}^j_k$.

Fact 4: Parameter $\theta^j_{km}$ can only appear in $\tilde{p}^j_k$ and cannot appear in $\tilde{p}^j_k$ or $\tilde{p}^j_j$ for $j \neq k$.

Proof: This is a direct result of Assumption 1.

As a direct consequence of Fact 4, we see that

$$
\frac{\partial f^j}{\partial \theta^j_{km}} = \begin{cases} 
\frac{\partial f^j}{\partial \theta^j_{km}} & \text{if } j \neq 1 \\
0 & \text{if } j \neq 1
\end{cases}
$$

Hence, (15) and (16) simplify to

$$
\frac{\partial y_k}{\partial \theta^j_{km}} = \frac{\partial y_k}{\partial \theta^j_{km}} + \frac{\partial y_k}{\partial \theta^j_{km}}
$$

$$(23)$$

$$(24)$$

so that [see (14)]

$$
\frac{\partial y_k}{\partial \theta^j_{km}} + \frac{\partial y_k}{\partial \theta^j_{km}} = \left( \frac{\partial y_k}{\partial \theta^j_{km}} + \frac{\partial y_k}{\partial \theta^j_{km}} \right) + \left( \frac{\partial y_k}{\partial \theta^j_{km}} + \frac{\partial y_k}{\partial \theta^j_{km}} \right) \left( \frac{\partial \tilde{\mu}^j_k(x_k) / \partial \theta^j_{km}}{\partial \tilde{\mu}^j_k(x_k) / \partial \theta^j_{km}} \right).
$$

(27)

Fact 5: It is true that

$$
\frac{\partial \tilde{\mu}^i_k(x_k) / \partial \theta^j_{km}}{\partial \tilde{\mu}^i_k(x_k) / \partial \theta^j_{km}} = \frac{1}{2} \left[ f_{\varepsilon^2} (x^2) - y^2 \right] \left[ \frac{\partial y_k}{\partial \theta^j_{km}} + \frac{\partial y_k}{\partial \theta^j_{km}} \right] \left( \frac{\partial \tilde{\mu}^j_k(x_k) / \partial \theta^j_{km}}{\partial \tilde{\mu}^j_k(x_k) / \partial \theta^j_{km}} \right).
$$

(28)

9 We hasten to point out that if an MF parameter is shared across some or all antecedent MFs, then (23) and (24) are invalid, and a different analysis must be performed from this point on.

10 For minimum t-norm, note for example, that $\min_p [\alpha(\theta_1), \beta(\theta_2), \gamma(\theta_3)] = \min_p [\alpha(\theta_1), \beta(\theta_2), \gamma(\theta_3)]$.

For product t-norm,

$$
\frac{\partial f^j}{\partial \theta^j_{km}} = \frac{\partial f^j}{\partial \theta^j_{km}} \times \frac{\partial \tilde{\mu}^j_k(x_k)}{\partial \theta^j_{km}}(x_k)
$$

(29a)

and for minimum t-norm

$$
\frac{\partial f^j}{\partial \theta^j_{km}} = \frac{\partial f^j}{\partial \theta^j_{km}} \times \frac{\partial \tilde{\mu}^j_k(x_k)}{\partial \theta^j_{km}}(x_k)
$$

(29b)

where $\min_p \{ x \} = 1$, when $\min_p \{ x \} = 0$, and $\min_p \{ x \} = 0$. When $\min_p \{ x \} = 0$. When $\min_p \{ x \} = 0$.

Proof: Apply the chain rule to (A-11) and (A-12) making use of Fact 4.

The remaining calculations of $\partial \tilde{\mu}^j_k(x_k) / \partial \theta^j_{km}$ and $\partial \tilde{\mu}^j_k(x_k) / \partial \theta^j_{km}$ require specification of antecedent MFs and their associated FOU.

So, as to see the forest from the trees, we next present a computational flow chart (see Fig. 2) for the calculations of $\partial \tilde{\mu}^j_k(x_k) / \partial \theta^j_{km}$. We show the inherent parallelism in the computations and where additional information is needed before the computations can be completed. Many of the computations only have to be done one time regardless of which antecedent parameters are tuned. The ones in the heavier outlined blocks must be done for each $\theta^j_{km}$.

See Appendix B for comparable results for an interval type-2 nonsingleton type-2 FLS.
TABLE VI
MEANINGS OF $y^i_1$, $y^i_2$, $\mathcal{f}^i$, $\overline{\mathcal{f}}^i$ and $M$ in (A-15) for Different Type-Reduction Methods\textsuperscript{a}. In This Table, All Symbols Refer to Rule-Ordered Quantities

<table>
<thead>
<tr>
<th>Type-reduction method</th>
<th>$y^i_1$ and $y^i_2$ defined</th>
<th>$\mathcal{f}^i$ and $\overline{\mathcal{f}}^i$ defined\textsuperscript{d}</th>
<th>$M$ defined</th>
</tr>
</thead>
<tbody>
<tr>
<td>center-of-sets</td>
<td>left and right end points of the centroid of the consequent of the $i$th rule; treated as consequent parameters</td>
<td>lower and upper firing degrees of the $i$th rule; contains antecedent MF parameters</td>
<td>number of rules</td>
</tr>
<tr>
<td>centroid\textsuperscript{b}</td>
<td>$y^i_1 = y^i_2 = y^i$, the $i$th point in the sampled universe of discourse of the FLS’s output</td>
<td>lower and upper membership grades of the $i$th sampled output of the FLS’s output; contains antecedent and consequent MF parameters</td>
<td>number of sampled points</td>
</tr>
<tr>
<td>center-of-sums\textsuperscript{c}</td>
<td>$y^i_1 = y^i_2 = y^i$, the $i$th point in the sampled universe of discourse of the FLS’s output</td>
<td>sums of lower and upper membership grades for the $i$th sampled point of all rule outputs; contains antecedent and consequent MF parameters</td>
<td>number of sampled points</td>
</tr>
<tr>
<td>height</td>
<td>$y^i_1 = y^i_2 = y^i$, a single point in the consequent domain of the $i$th rule, usually chosen to be the point having the highest primary membership in the principal MF of the output set; treated as a consequent parameter\textsuperscript{d}</td>
<td>lower and upper firing degrees of the $i$th rule; contains antecedent MF parameters</td>
<td>number of rules</td>
</tr>
</tbody>
</table>

\textsuperscript{a} Comparable results for modified height type-reduction can be found in [6] Section 9.5.4.

\textsuperscript{b} Prior to calculating the centroid type-reduced set, the fired type-2 fuzzy sets are unioned.

\textsuperscript{c} Prior to calculating the center-of-sums type-reduced set, the membership functions of the fired type-2 fuzzy sets are added (or a linear combination of them is formed).

\textsuperscript{d} See Section A.2 for formulas for $\mathcal{f}^i$ and $\overline{\mathcal{f}}^i$, as well as for rule outputs and unioned rule outputs.

V. CALCULATION OF $\frac{\partial \mathcal{f}^i}{\partial \theta^j}$ FOR CONSEQUENT PARAMETERS

Generally speaking, consequent parameters are the parameters that characterize consequent MFs. When, however, we use center-of-set type-reduction, as we have assumed in Assumption 4 (Section II), then those parameters can be replaced by the two end-points of the centroids of the type-2 consequent sets (see Table A-I in Appendix A). Doing this can reduce the number of design parameters. For example, if the consequent MF is also a Gaussian primary MF with uncertain mean, that is characterized by three design parameters, then using the two end-points of the centroids of this type-2 MF reduces the number of design parameters from three to two.

Note that the consequent parameters do not need the "$k" or "$m" subscripts in $\theta^i_{k,m}$ ($k$ and $m$ are associated with a specific antecedent). Additionally, (see the formulas for $y^i_1$ and $y^i_2$ in Table I) $\theta^i = z_1^i$ or $\theta^i = z_2^i$.\textsuperscript{11}

\textsuperscript{11} Note that this parameterization is true for center-of-sets and height type-reduction but is not true for centroid or center of sums type-reduction.
From (2) and (13), it is easy to show that
\[
\frac{\partial c^{(i)}}{\partial z^j_l} = \frac{\partial c^{(i)}}{\partial s_{x_{k2}}} \frac{\partial s_{x_{k2}}}{\partial x^{(i)}} \frac{\partial x^{(i)}}{\partial y} \frac{\partial y}{\partial z^j_l} = \frac{1}{2} \left[ f_{s_{x_{k2}}} \left( x^{(i)} \right) - y^{(i)} \right] \frac{\partial y}{\partial z^j_l}
\]
and
\[
\frac{\partial c^{(i)}}{\partial z^j_l} = \frac{1}{2} \left[ f_{s_{x_{k2}}} \left( x^{(i)} \right) - y^{(i)} \right] \frac{\partial y}{\partial z^j_l}.
\]

**Fact 7:** It is true that
\[
\frac{\partial y}{\partial z^j_l} = e_j^T \left( \frac{M_{x_{k1}}^T + M_{x_{k2}}^T f}{r_r^T Q_r + s_l^T Q_l} \right)
\]
\[
\frac{\partial y}{\partial z^j_l} = e_j^T \left( \frac{M_{x_{k1}}^T + M_{x_{k2}}^T f}{r_r^T Q_r + s_l^T Q_l} \right)
\]
where \( e_j \) is the \( j \)th \( M \times 1 \) unit vector.

**Proof:** Using the vector calculus fact that \( \nabla z^T y = \alpha \), it is easy to show, from (7) and (12), that
\[
\nabla y^T y = \left( \frac{f^T M_{x_{k1}} + f^T M_{x_{k2}}}{r_r^T Q_r + s_l^T Q_l} \right)^T
\]
\[
= \left( \frac{M_{x_{k1}}^T f + M_{x_{k2}}^T f}{r_r^T Q_r + s_l^T Q_l} \right)
\]

Equations (33) and (34) follow directly upon application of \( e_j^T \) to (35) and (36), respectively.\(^\text{12}\)

\( \text{12Recall that } \nabla y^T y = \left[ \partial y/\partial z^j_1, \partial y/\partial z^j_2, \ldots, \partial y/\partial z^j_l \right]^T, \text{ so that } e_j^T \nabla y^T y = \partial y/\partial z^j_l. \)
TABLE VIII
DEFINITIONS OF THE FIVE STATES. FOR MINIMUM T-NORM, REPLACE $\sigma_k^2$ BY $\sigma_k$, $\sigma_k'$ BY $\sigma_k$ AND $\sigma_l^2$ BY $\sigma_l$.

<table>
<thead>
<tr>
<th>State</th>
<th>Product t-norm</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>$x'_i \leq m^l_i$ and $x'_i &lt; \frac{m^l_i + m^u_i}{2} - \frac{\sigma^2_i (m^u_i - m^l_i)}{2\sigma^2_k}$</td>
</tr>
<tr>
<td>(2)</td>
<td>$x'_i \in [m^l_i, m^u_i]$ and $x'_i &lt; \frac{m^l_i + m^u_i}{2} - \frac{\sigma^2_i (m^u_i - m^l_i)}{2\sigma^2_k}$</td>
</tr>
<tr>
<td>(3)</td>
<td>$x'_i \in [m^l_i, m^u_i]$ and $x'_i \in \left[ \frac{m^l_i + m^u_i}{2} - \frac{\sigma^2_i (m^u_i - m^l_i)}{2\sigma^2_k}, \frac{m^l_i + m^u_i}{2} + \frac{\sigma^2_i (m^u_i - m^l_i)}{2\sigma^2_k} \right]$</td>
</tr>
<tr>
<td>(4)</td>
<td>$x'_i \in [m^l_i, m^u_i]$ and $x'_i &gt; \frac{m^l_i + m^u_i}{2} + \frac{\sigma^2_i (m^u_i - m^l_i)}{2\sigma^2_k}$</td>
</tr>
<tr>
<td>(5)</td>
<td>$x'_i \geq m^u_i$ and $x'_i &gt; \frac{m^l_i + m^u_i}{2} + \frac{\sigma^2_i (m^u_i - m^l_i)}{2\sigma^2_k}$</td>
</tr>
</tbody>
</table>

This completes the derivations of general derivative formulas. To proceed further, the FOUs of MFs must be specified. Because calculations of $\partial f_i^{(l)}/\partial \theta_k$ for consequent parameters are so straightforward, we do not include a flowchart for their implementation. Just implement (31)–(34), and use the vectors and matrices that are defined in Table II.

VI. EXAMPLE

In order to compute $\frac{\partial f_i}{\partial \theta_k}$ and $\frac{\partial^2 f_i}{\partial \theta_k^2}$ using (39a) and (30a), we need $\frac{\partial f_i^{(l)}(x_k)}{\partial \theta_k}$ and $\frac{\partial f_i^{(l)}(x_k)}{\partial \theta_k^2}$. Here, we will compute $\frac{\partial f_i^{(l)}(x_k)}{\partial \theta_k}$ and $\frac{\partial f_i^{(l)}(x_k)}{\partial \theta_k^2}$ for Gaussian primary MFs with uncertain means (see Example A-1). To start off, we restate the results of Example A-1 using the more explicit notations for antecedent MFs ($k = 1, \ldots, p$ and $l = 1, \ldots, M$)

$$\mu^l_k(x_k) = \exp \left\{ -\frac{1}{2} \left( \frac{x_k - m^l_k}{\sigma^l_k} \right)^2 \right\} \frac{m^l_k}{m_{21}^l, m_{22}^l} \begin{cases} 1 & x_k < m^l_k \\ \frac{N(m_{21}^l, \sigma_1^l, m_{22}^l)}{x_k \leq m^l_k} & m^l_k \leq x_k \leq m^l_k \\ \frac{N(m_{21}^l, \sigma_1^l, m_{22}^l)}{x_k > m^l_k} & \end{cases} \tag{37}$$

$$\tilde{\mu}^l_k(x_k) = \begin{cases} N(m_{21}^l, \sigma_1^l, m_{22}^l) & x_k < m^l_k \\ 1 & m^l_k \leq x_k \leq m^l_k \\ 0 & x_k > m^l_k \end{cases} \tag{38}$$

$$\nu^l_k(x_k) = \begin{cases} N(m_{21}^l, \sigma_1^l, m_{22}^l) & x_k \leq \frac{(m^l_k + m^u_k)}{2} \\ \frac{N(m_{21}^l, \sigma_1^l, m_{22}^l)}{x_k > \frac{(m^l_k + m^u_k)}{2}} \end{cases} \tag{39}$$

We summarize the parameters that $\mu^l_k(x_k)$ and $\nu^l_k(x_k)$ depend upon, as a function of $x_k$, in Table III. Its results were obtained by examining (38) and (39) (see, also, Fig. 3). Let $\theta_{k,1}^l \equiv m_{21}^l, \theta_{k,2}^l \equiv m_{22}^l$, and $\theta_{k,3}^l \equiv \sigma_1^l$. Tables IV and V provide $\frac{\partial f_i^{(l)}(x_k)}{\partial \theta_k}$ and $\frac{\partial f_i^{(l)}(x_k)}{\partial \theta_k^2}$, respectively.

The results in these Tables IV and V would be used in Fig. 3 as follows. The tests on variable $x_k$, given in Tables IV and V, let us implement the top two blocks in which we have to determine the active branches of the lower and upper MFs. The results in Table V provide $\frac{\partial f_i^{(l)}(x_k)}{\partial \theta_k}$ which is needed to compute $\frac{\partial f_i^{(l)}(x_k)}{\partial \theta_k}$, and the results in Table IV provide $\frac{\partial f_i^{(l)}(x_k)}{\partial \theta_k^2}$ which is needed to compute $\frac{\partial^2 f_i^{(l)}(x_k)}{\partial \theta_k^2}$.

TABLE IX
UPPER AND LOWER MF PARAMETER DEPENDENCIES (SEE TABLE VII)

<table>
<thead>
<tr>
<th>State</th>
<th>$\mu^{\text{up}}<em>{l}(x</em>{k,\text{max}})$</th>
<th>$\mu^{\text{lo}}<em>{l}(x</em>{k,\text{max}})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>$m^l_{21}, \sigma^l_{l}, \sigma_{k_1}$</td>
<td>$m^l_{21}, \sigma^l_{l}, \sigma_{k_1}$</td>
</tr>
<tr>
<td>(2)</td>
<td>$m^l_{21}, \sigma^l_{l}, \sigma_{k_1}$</td>
<td>None</td>
</tr>
<tr>
<td>(3)</td>
<td>$m^l_{21}, \sigma^l_{l}, \sigma_{k_1}$</td>
<td>None</td>
</tr>
<tr>
<td>(4)</td>
<td>$m^l_{21}, \sigma^l_{l}, \sigma_{k_1}$</td>
<td>None</td>
</tr>
<tr>
<td>(5)</td>
<td>$m^l_{21}, \sigma^l_{l}, \sigma_{k_1}$</td>
<td>None</td>
</tr>
</tbody>
</table>

* Includes input MF parameter dependencies.

We have made type-2 FLSs much more accessible to FLS designers by providing mathematical formulas and computational flowcharts for computing the derivatives that are needed to implement steepest-descent parameter tuning algorithms for such systems. We have demonstrated why computing such derivatives is much more challenging than it is for a type-1 FLS, and have provided derivative calculations that are applicable to any kind of type-2 MF, since most of the calculations can be performed without prespecifying the nature of those MFs. Eventually, one does have to specify the nature of the type-2 MF in order to complete the calculations. We showed how to complete the calculations for a Gaussian primary MF with uncertain means.

It is important for the reader to remember the four assumptions stated in Section II. If any of them are not obeyed, then some or all of the results of this paper must be modified.
TABLE X
DERIVATIVES OF
\(\mu_{\delta_k}^\ast (\delta_{\text{max}}) \equiv u\)

<table>
<thead>
<tr>
<th>State</th>
<th>(\partial \mu/\partial \theta_{k,1}^i)</th>
<th>(\partial \mu/\partial \theta_{k,2}^i)</th>
<th>(\partial \mu/\partial \theta_{k,3}^i)</th>
<th>(\partial \mu/\partial \theta_{k,4}^i)</th>
<th>(\partial \mu/\partial \theta_{k,5}^i)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>0</td>
<td>(\partial(1)/\partial \theta_{k,2}^i)</td>
<td>(\partial(1)/\partial \theta_{k,3}^i)</td>
<td>(\partial(1)/\partial \theta_{k,4}^i)</td>
<td>0</td>
</tr>
<tr>
<td>(2)</td>
<td>0</td>
<td>(\partial(1)/\partial \theta_{k,2}^i)</td>
<td>(\partial(1)/\partial \theta_{k,3}^i)</td>
<td>(\partial(1)/\partial \theta_{k,4}^i)</td>
<td>0</td>
</tr>
<tr>
<td>(3)</td>
<td>(\partial(3)/\partial \theta_{k,1}^i)</td>
<td>(\partial(3)/\partial \theta_{k,2}^i)</td>
<td>(\partial(3)/\partial \theta_{k,3}^i)</td>
<td>(\partial(3)/\partial \theta_{k,4}^i)</td>
<td>0</td>
</tr>
<tr>
<td>(4)</td>
<td>(\partial(4)/\partial \theta_{k,1}^i)</td>
<td>(\partial(4)/\partial \theta_{k,2}^i)</td>
<td>(\partial(4)/\partial \theta_{k,3}^i)</td>
<td>(\partial(4)/\partial \theta_{k,4}^i)</td>
<td>0</td>
</tr>
<tr>
<td>(5)</td>
<td>(\partial(4)/\partial \theta_{k,1}^i)</td>
<td>(\partial(4)/\partial \theta_{k,2}^i)</td>
<td>(\partial(4)/\partial \theta_{k,3}^i)</td>
<td>(\partial(4)/\partial \theta_{k,4}^i)</td>
<td>0</td>
</tr>
</tbody>
</table>

* e.g., \(\partial(4)/\partial \theta_{k,1}^i = \partial \mu/\partial \theta_{k,4}^i\) for State (4). Use Table VII to complete this computation.

TABLE XI
DERIVATIVES OF \(\pi_{\delta_k}^\ast (\pi_{\text{max}}) \equiv p\)

<table>
<thead>
<tr>
<th>State</th>
<th>(\partial \bar{\mu}/\partial \theta_{k,1}^i)</th>
<th>(\partial \bar{\mu}/\partial \theta_{k,2}^i)</th>
<th>(\partial \bar{\mu}/\partial \theta_{k,3}^i)</th>
<th>(\partial \bar{\mu}/\partial \theta_{k,4}^i)</th>
<th>(\partial \bar{\mu}/\partial \theta_{k,5}^i)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>(\partial(1)/\partial \theta_{k,1}^i)</td>
<td>0</td>
<td>(\partial(1)/\partial \theta_{k,3}^i)</td>
<td>0</td>
<td>(\partial(1)/\partial \theta_{k,5}^i)</td>
</tr>
<tr>
<td>(2)</td>
<td>0</td>
<td>0</td>
<td>(\partial(1)/\partial \theta_{k,3}^i)</td>
<td>0</td>
<td>(\partial(1)/\partial \theta_{k,5}^i)</td>
</tr>
<tr>
<td>(3)</td>
<td>0</td>
<td>0</td>
<td>(\partial(1)/\partial \theta_{k,3}^i)</td>
<td>0</td>
<td>(\partial(1)/\partial \theta_{k,5}^i)</td>
</tr>
<tr>
<td>(4)</td>
<td>0</td>
<td>0</td>
<td>(\partial(1)/\partial \theta_{k,3}^i)</td>
<td>0</td>
<td>(\partial(1)/\partial \theta_{k,5}^i)</td>
</tr>
<tr>
<td>(5)</td>
<td>(\partial(5)/\partial \theta_{k,1}^i)</td>
<td>(\partial(5)/\partial \theta_{k,2}^i)</td>
<td>(\partial(5)/\partial \theta_{k,3}^i)</td>
<td>0</td>
<td>(\partial(5)/\partial \theta_{k,5}^i)</td>
</tr>
</tbody>
</table>

* e.g., \(\partial(1)/\partial \theta_{k,1}^i = \partial \bar{\mu}/\partial \theta_{k,1}^i\) for State (1). Use Table VII to complete this computation.

APPENDIX A
CALCULATIONS REQUIRED TO IMPLEMENT AN INTERVAL SINGLETON TYPE-2 FUZZY LOGIC SYSTEM

In this appendix we summarize all of the calculations that are needed to implement an interval singleton type-2 FLS. For detailed derivations of all of these results as well as more background on type-2 FLSs, see [6]. To begin we define important terms that are associated with a type-2 fuzzy set and a type-2 FLS.

A. Preliminaries

Consider a type-2 FLS having \(p\) inputs \(x_1 \in X_1, \ldots, x_p \in X_p\), and one output \(y \in Y\). We assume there are \(M\) rules where the \(lth\) rule has the form

\[ R^l: \text{IF } x_1 \text{ is } \tilde{F}_1^l \text{ and } \ldots \text{ and } x_p \text{ is } \tilde{F}_p^l, \text{ THEN } y \text{ is } \tilde{C}_l^l, \quad l = 1, \ldots, M. \] \hfill (A-1)

This rule represents a type-2 relation between the input space \(X_1 \times \ldots \times X_p\), and the output space, \(Y\), of the type-2 FLS. Associated with the \(p\) antecedent type-2 fuzzy sets, \(\tilde{F}_1^l, \ldots, \tilde{F}_p^l\), are the type-2 MFs \(\mu_{\tilde{F}_i}^l (i = 1, \ldots, p)\), and associated with the consequent type-2 fuzzy set \(\tilde{C}_l^l\) is its type-2 MF \(\mu_{\tilde{C}_l}^l\). We frequently use the simpler notation \(\hat{A}\) for \(\mu_{\tilde{A}}\).  

Definition A-1: A type-2 fuzzy set, denoted \(\hat{A}\), is characterized by a (three-dimensional) type-2 membership function (MF) \(\mu_{\hat{A}}(x, u)\), i.e.,

\[ \hat{A} = \int \int \mu_{\hat{A}}(x, u)/(x, u) \quad J_x \subseteq [0, 1] \] \hfill (A-2)

where \(\int \int \) denotes union over all admissible \(x\) and \(u\), and \(0 \leq \mu_{\hat{A}}(x, u) \leq 1\). At each fixed value of \(x \in X\), \(J_x\) is the primary membership of \(x\) and \(x\) is called the primary variable. If \(\mu_{\hat{A}}(x, u) \equiv 1\), then \(\mu_{\hat{A}}(x, u)\) is a vertical slice of \(\mu_{\hat{A}}(x, u)\), and \(\mu_{\hat{A}}(x, u)\) is a vertical slice of its MF \(\mu_{\hat{A}}(x, u)\). A secondary MF is a vertical slice of \(\mu_{\hat{A}}(x, u)\) which we also refer to as a secondary set.

Definition A-2: The domain of a secondary MF \(\mu_{\hat{A}}(x, u)\) is called the primary membership of \(x\). In (A-2) and (A-3), \(J_x\) is the primary membership of \(x\), where \(J_x \subseteq [0, 1]\) for \(\forall x \in X\).

Definition A-3: The amplitude of a secondary MF \(\mu_{\hat{A}}(x, u)\) is called a secondary grade. In (A-3), \(f_{\mu_{\hat{A}}}(u)\) is a secondary grade.

Definition A-4: An interval type-2 fuzzy set is a type-2 fuzzy set all of whose secondary MFs are type-1 interval sets, i.e., \(f_{\mu_{\hat{A}}}(u) = 1, \forall u \in J_x \subseteq [0, 1]\) and \(\forall x \in X\).

Definition A-5: An interval type-2 fuzzy set is a type-2 fuzzy set all of whose secondary MFs are type-1 interval sets, i.e., \(f_{\mu_{\hat{A}}}(u) = 1, \forall u \in J_x \subseteq [0, 1]\) and \(\forall x \in X\).

Interval secondary MFs reflect a uniform uncertainty at the primary memberships of \(x\), and are the ones most commonly used in a type-2 FLS. Note that an interval set can be represented just by its domain interval, which can be expressed in terms of its left and right end-points as \([a, b]\), or by its center and spread as \([c - s, c + s]\), where \(c = (l + r)/2\) and \(s = (r - l)/2\).
Definition A-6: Uncertainty in the primary memberships of a type-2 fuzzy set, \( \tilde{A} \), consists of a bounded region that we call the FOU. It is the union of all primary memberships, i.e.,

\[
\text{FOU}(\tilde{A}) = \bigcup_{x \in X} J_x. \quad (A-4)
\]

The term FOU is very useful, because it not only focuses our attention on the uncertainties inherent in a specific type-2 MF, whose shape is a direct consequence of the nature of these uncertainties, but it also provides a very convenient verbal description of the entire domain of support for all the secondary grades of a type-2 MF.

Definition A-7: Consider a family of type-1 MFs \( \mu_A(x|p_1, p_2, \ldots, p_v) \) where \( p_1, p_2, \ldots, p_v \) are parameters, some or all of which vary over some range of values, i.e., \( p_i \in P_i(i = 1, \ldots, v) \). A primary MF is any one of these type-1 MFs, e.g., \( \mu_A(x|p_1 = p_1, p_2 = p_2, \ldots, p_v = p_v) \). For short, we use \( \mu_A(x) \) to denote a primary MF. It will be subject to some restrictions on its parameters. The family of all primary MFs creates a FOU.

Two examples of very useful primary MFs are: Gaussian MF with uncertain mean and certain standard deviation, and Gaussian MF with certain mean and uncertain standard deviation.

Definition A-8: An upper MF and a lower MF are two type-1 MFs that are bounds for the FOU of a type-2 fuzzy set \( \tilde{A} \). The upper MF is associated with the upper bound of FOU(\( \tilde{A} \)), and is denoted \( \overline{\mu}_A(x) \), \( \forall x \in X \). The lower MF is associated with the lower bound of FOU(\( \tilde{A} \)), and is denoted \( \underline{\mu}_A(x) \), \( \forall x \in X \), i.e.,

\[
\overline{\mu}_A(x) = \text{FOU}(\tilde{A}) \quad \forall x \in X \quad (A-5)
\]

and

\[
\underline{\mu}_A(x) = \text{FOU}(\tilde{A}) \quad \forall x \in X \quad (A-6)
\]

Because the domain of a secondary MF has been constrained in Definition A-1 to be contained in \([0, 1]\), lower and upper MFs always exist.

Example A-1: Consider the case of a Gaussian primary MF having a fixed standard deviation, \( \sigma \), and an uncertain mean that takes on values in \([m_1, m_2]\), i.e.,

\[
\mu_A(x) = \exp \left[ -\frac{1}{2} \left( \frac{x - m}{\sigma} \right)^2 \right] \quad m \in [m_1, m_2]. \quad (A-7)
\]

Corresponding to each value of \( m \) we will get a different membership curve (Fig. 3). The uniform shading for the FOU denotes interval sets for the secondary MFs and represents the entire interval type-2 fuzzy set \( \mu_A(x, u) \).

The upper MF, \( \overline{\mu}_A(x) \), is

\[
\overline{\mu}_A(x) = \begin{cases} 
N(m_1, \sigma; x) & x < m_1 \\
1 & m_1 \leq x \leq m_2 \\
N(m_2, \sigma; x) & x > m_2
\end{cases} \quad (A-8)
\]

where, for example, \( N(m_1, \sigma; x) \equiv \exp\left[-(1/2)(x - m_1/\sigma)^2\right] \).

The thick solid curve in Fig. 3 denotes the upper MF. The lower MF, \( \underline{\mu}_A(x) \), is

\[
\underline{\mu}_A(x) = \begin{cases} 
N(m_2, \sigma; x) & x \leq m_1 + m_2 \\
N(m_1, \sigma; x) & x > m_1 + m_2/2
\end{cases} \quad (A-9)
\]

The thick dashed curve in Fig. 3 denotes the lower MF.

From this example we see that sometimes an upper (or a lower) MF cannot be represented by just one mathematical function over its entire \( x \)-domain. It may consist of several branches each defined over a different segment of the entire \( x \)-domain.\(^{13}\)

When the input, \( x \), is located in a specific \( x \)-domain segment, we call its corresponding MF branch an active branch \([4], [6]\), e.g., in (A-9), when \( x > (m_1 + m_2)/2 \), the active branch for \( \underline{\mu}_A(x) \) is \( N(m_2, \sigma; x) \).

B. Fuzzy Inference Engine Results

The major result for an interval singleton type-2 FLS is summarized in the following.

Theorem A-1: [5], [6] In an interval singleton type-2 FLS using product or minimum t-norm: a) the result of the input and antecedent operations, is an interval type-1 set, called the firing set, \( I \),

\[
F^I(\chi') = \left[ \overline{f}^I(\chi'), \underline{f}^I(\chi') \right] = \left[ \overline{f}_1, \underline{f}_1 \right] \quad (A-10)
\]

where

\[
\overline{f}^I(\chi') = \mu_{\overline{B}_1}(x_1') \cdots \mu_{\overline{B}_p}(x_p') = \mu_{\overline{B}_1}(x_1) \cdots \mu_{\overline{B}_p}(x_p) \quad (A-11)
\]

and

\[
\underline{f}^I(\chi') = \mu_{\underline{B}_1}(x_1') \cdots \mu_{\underline{B}_p}(x_p') = \mu_{\underline{B}_1}(x_1) \cdots \mu_{\underline{B}_p}(x_p) \quad (A-12)
\]

b) the rule \( R^f \) fired output consequent set, \( \mu_{\overline{B}_y}(y) \), is the interval type-2 fuzzy set

\[
\mu_{\overline{B}_y}(y) = \int_{\chi \in \left[ \underline{f}_1, \overline{f}_1 \right]} 1/|H_y|, \quad y \in Y \quad (A-13)
\]

where \( \mu_{\overline{B}_y}(y) \) and \( \mu_{\underline{B}_y}(y) \) are the lower and upper membership grades of \( \mu_{\overline{B}_y}(y) \). (c) Suppose that \( N \) of the \( M \) rules in the FLS fire, where \( N \leq M \), and the combined output type-1 fuzzy set, \( \mu_{\overline{B}_y}(y) \), is obtained by combining the fired output consequent sets by taking the union of the rule \( R^f \) fired output consequent sets; then (A-14), as shown at the bottom of the next page, holds.

A complete proof of this theorem can be found in [5] and [6]. Generalizations of this theorem to the very important case when the input to the type-2 FLS is a type-2 fuzzy set are also given in those references (see, also Appendix B for some of those results).

C. Type-Reduction

Five different type-reduction methods are described in [6]. Each is inspired by what we do in a type-1 FLS, when we defuzzify the (combined) output of the inference engine using a va-
riety of defuzzification methods that all do some sort of centroid calculation, and, is based on computing the centroid of a type-2 fuzzy set. Using the Extension Principle, Karnik and Mendel [2] defined the centroid of a type-2 fuzzy set; it is a type-1 fuzzy set. Computing the centroid of a general type-2 fuzzy set can be very intensive; however, for an interval type-2 fuzzy set, two exact iterative methods for computing its centroid have been developed by them. This was possible because the centroid of an interval type-2 fuzzy set is an interval type-1 fuzzy set, and such sets are completely characterized by their left- and right-end points; hence, computing the centroid of an interval type-2 fuzzy set only requires computing those two end-points.

The different kinds of type-reduction can all be expressed as

$$\begin{align*}
Y_{TR}(x) = [y_l, y_r] = & \int_{y_l \in [y_{L1}, y_{L2}]} \cdots \int_{y_r \in [y_{R1}, y_{R2}]} \int_{f_{IL} \in [f_{I}^{M}]} \int_{f_{IR} \in [f_{I}^{M}]} \\
& \cdots \int_{f_{R2} \in [f_{R}^{M}]} \int_{f_{R1} \in [f_{R}^{M}]} 1/\sum_{i=1}^{M} f_{i}^{M} g_{i}^{M} f_{j}^{M}^{I} f_{j}^{M}^{R} (A-15) \end{align*}$$

in which $y_{L1}, y_{L2}, f_{I}^{1}, f_{I}^{2}$, and $M$ have different meanings, as summarized in Table VI. In this paper, we only focus on center-of-sets type-reduction.

In (A-15), all symbols refer to quantities that are rule-ordered. In the Karnik–Mendel iterative procedures, (one for computing $y_l$ and one for computing $y_r$) that we summarize next, all quantities are reordered according to step 1: The Karnik–Mendel iterative procedure for computing $y_{T}$ is as follows:14

1) Without loss of generality, assume that the precomputed $y_{i}^{L}$ are arranged in ascending order; i.e., $y_{1}^{L} \leq y_{2}^{L} \leq \cdots \leq y_{M}^{L}$. Re-order the $f_{i}^{M}$ accordingly and call them $f_{i}^{L}$.

2) Compute $y_{L}$ as $y_{L} = \sum_{i=1}^{M} g_{i}^{L} f_{i}^{L} / \sum_{i=1}^{M} g_{i}^{L}$, by initially setting $g_{i}^{L} = (g_{i}^{L} + \bar{g}_{i}^{L}) / 2$ for $i = 1, \ldots, M$, where $g_{i}^{L}$ and $\bar{g}_{i}^{L}$ have been previously computed using (A-11) and (A-12), respectively, and let $y_{L}^{0} \equiv y_{L}$.

3) Find $R(1 \leq R \leq M - 1)$ such that $y_{R}^{0} \geq y_{L}^{0} \neq y_{R+1}^{0}$.

4) Compute $y_{L} = \sum_{i=1}^{M} g_{i}^{L} f_{i}^{L} / \sum_{i=1}^{M} g_{i}^{L}$ with $g_{i}^{L} = g_{i}^{L}$ for $i = 1, \ldots, R$, and set $y_{L}^{R} = y_{L}^{0}$. For $i > R$, and set $y_{R+1}^{0} = y_{L}^{R}$. If $y_{L}^{R} = y_{L}^{R}$, then stop and set $y_{L}^{R} = y_{L}^{R}$.

6) Set $y_{L}^{R+1}$ equal to $y_{L}^{R+1}$, and return to Step 3.

The iterative procedure for computing $y_{R}$ is very similar to the one just given for $y_{L}$. In step 1, it is now the precomputed $y_{i}^{R}$ that are arranged in ascending order; i.e., $y_{1}^{R} \leq y_{2}^{R} \leq \cdots \leq y_{M}^{R}$, and the $f_{i}^{R}$ that are reordered accordingly (they are now called $g_{i}^{R}$). In step 2, $y_{R}$ is computed as $y_{R} = \sum_{i=1}^{M} g_{i}^{R} f_{i}^{R} / \sum_{i=1}^{M} g_{i}^{R}$ by initially setting $g_{i}^{R} = (g_{i}^{R} + \bar{g}_{i}^{R}) / 2$ for $i = 1, \ldots, M$. In step 3, $L(1 \leq L \leq M - 1)$ is found such that $y_{L}^{R} \leq y_{L}^{R} \leq y_{L+1}^{R}$, and, in step 4, $y_{L}$ is computed as $y_{L} = \sum_{i=1}^{M} g_{i}^{L} f_{i}^{L} / \sum_{i=1}^{M} g_{i}^{L}$ with $g_{i}^{L} = \bar{g}_{i}^{L}$ for $i \leq L$ and $g_{i}^{L} = g_{i}^{L}$ for $i > L$.

These two four-step iterative procedures (steps 1 and 2) are initializations steps have been proven by Karnik and Mendel [2] to converge to the exact solutions in no more than $M$ iterations.

Observe that in these procedures, the computed numbers $R$ and $L$ (called cross-over points or switch-points) are very important. For $i \leq R$, $g_{i}^{L} = g_{i}^{L}$, whereas for $i > R$ $g_{i}^{L} = \bar{g}_{i}^{L}$; hence, $y_{L}$ can be represented as

$$y_{L} = y_{L} (g_{L}^{1} \cdots, g_{L}^{R}, g_{L}^{R+1} \cdots, g_{L}^{M}, y_{L}^{1} \cdots, y_{L}^{R}) . \quad (A-16)$$

Additionally, $g_{i}^{L} = \bar{g}_{i}^{L}$ for $i \leq L$ and $g_{i}^{L} = \bar{g}_{i}^{L}$ for $i > L$, so that $y_{L}$ can be represented as

$$y_{L} = y_{L} (g_{L}^{1} \cdots, g_{L}^{L}, g_{L}^{L+1} \cdots, g_{L}^{M}, y_{L}^{1} \cdots, y_{L}^{L}) . \quad (A-17)$$

D. Defuzzification

Because $Y_{TR}$ is an interval set, we defuzzify it using the average of $y_{L}$ and $y_{R}$; hence, the defuzzified output of an interval singleton type-2 FLS is

$$y(x) = f_{o}c(x) = \frac{y_{L} + y_{R}}{2} . \quad (A-18)$$

APPENDIX B

RESULTS FOR INTERVAL TYPE-2 NONSINGLETOR TYPE-2 FLSs

In this appendix, we provide results comparable to those given in the main body of the paper, but for an interval type-2 nonsingleton type-2 FLS. For such a FLS not only are the rule antecedents and consequent characterized by interval type-2 fuzzy sets, but the inputs that activate the FLS are also interval type-2 fuzzy sets (a special case of which is a type-1 fuzzy set). We denote the MF for input $x_{i}$ by $\mu_{X_{i}}(x_{i})$, with lower and upper MFs $\mu_{X_{i}}^{L}(x_{i})$ and $\mu_{X_{i}}^{R}(x_{i})$, respectively.

A. Fuzzy Inference Engine Results

The major results for an interval type-2 nonsingleton type-2 FLS are in [5] and are also summarized in [6, Th. 12–11]. For the purposes of this paper, we only need the following procedure to compute $F^{L}(x') = [F^{L}(x'), F^{R}(x')] \equiv [a^{L}, a^{R}]$.

1) Choose a t-norm (product or minimum) and create the functions $\mu_{X_{i}}^{L}(x_{i})$ and $\mu_{X_{i}}^{R}(x_{i})$, where

$$\mu_{X_{i}}^{L}(x_{i}) = \int_{x_{i} \in X_{i}} \mu_{X_{i}}^{L}(x_{i}) \mu_{X_{i}}^{R}(x_{i}) / x_{i} . \quad (B-1)$$

\[ \text{mu}_{\tilde{b}}(y) = \frac{1}{b} \int_{\mu_{\tilde{b}}(y) = 1}^{y \in Y_{i}} \left[ \int_{f_{L}^{R} \in [f_{L}^{M}]} \int_{f_{R}^{R} \in [f_{R}^{M}]} \ldots \int_{f_{R}^{R} \in [f_{R}^{M}]} \right] \]
and
\[ n_{a_k}^{2} (x_k) = \frac{1}{x_k \in X_k} \left[ \mu_{a_k}^{2} (x_k) \ast \nu_{a_k}^{2} (x_k) \right] / x_k. \] (B-2)

2) Let \( a_k^{2} \) and \( a_k^{2} \) denote the values of \( x_k \) that are associated with \( \nu_{a_k}^{2} (x_k) \) and \( \mu_{a_k}^{2} (x_k) \), respectively. Compute \( a_k^{2} \) and \( a_k^{2} \).

3) Evaluate \( f_{a_k}^{2} (x_k) \) and \( f_{a_k}^{2} (x_k) \) where
\[ f_{a_k}^{2} (x_k) = \mu_{a_k}^{2} (x_k) \] (B-3)
and
\[ f_{a_k}^{2} (x_k) = \mu_{a_k}^{2} (a_k^{2} \). (B-4)

Note that \( a_k^{2} \) and \( a_k^{2} \) will depend upon measurement \( x_k \).

4) Compute \( f(a) \) and \( f(a) \) as
\[ f(a) = T_{a=1} f_{a_k}^{2} (x_k) = T_{a=1} \mu_{a_k}^{2} (a_k^{2} \] (B-5)
and
\[ f(a) = T_{a=1} f_{a_k}^{2} (x_k) = T_{a=1} \mu_{a_k}^{2} (a_k^{2} \]. (B-6)

Note that these results are comparable to those in part a) of Theorem A-1. The main difference between computing the firing interval for an interval type-2 nonsingleton type-2 FLS and an interval singleton type-2 FLS is having to compute \( a_k^{2} \) and \( a_k^{2} \). Note also that parts b) and c) of Theorem A-1 apply as is to the present nonsingleton case.

B. Changes

The changes in Sections I–V are as follows.

1) In Section I, change \( f_{a_k} \) in (2) to \( f_{a_k} \), where \( \mu_{a_k} = \mu_{a_k} \) is short for type-2 nonsingleton type-2.

2) In all sections, wherever the phrase “interval singleton type-2 FLS” is used, replace it with “interval type-2 nonsingleton type-2 FLS.”

3) There are no changes to Sections II and III.

4) Fact 4 is changed to: For specific values of \( k \) and \( i \), antecedent parameters \( \theta_{m,n} \) can only appear in \( \mu_{a_k}^{2} (a_k^{2} \) and \( \nu_{a_k}^{2} (a_k^{2} \) and cannot appear in \( \mu_{a_k}^{2} (a_k^{2} \) and \( \nu_{a_k}^{2} (a_k^{2} \) for \( j \neq k \). This is a direct result of Assumption 1 in Section II and the facts that \( a_k^{2} \) and \( a_k^{2} \) only depend upon one value of \( k \) and \( l \).

5) In Fact 5, change \( f_{a_k} \) in (28) to \( f_{a_k} \). In Section II, the facts that \( a_k^{2} \) and \( a_k^{2} \) only depend upon one value of \( k \) and \( l \).

6) In Fact 6, replace \( f_{a_k} (x_k) \) by \( f_{a_k} (a_k^{2} \) and \( f_{a_k} (x_k) \) by \( f_{a_k} (a_k^{2} \) for all \( i = 1, \ldots, M \). Although we will still refer to the equations for computing \( \partial f_{a_k} (x_k) / \partial \theta_{k,m} \) and \( \partial f_{a_k} (x_k) / \partial \theta_{k,m} \) as (29) and (30), respectively, it is to be understood that those equations are now modulo these changes.

7) The flowchart in Fig. 2 is modified as follows.

a. Replace the top four blocks of Fig. 2 by the blocks in Fig. 4.

b. Change \( f_{a_k} \) in the center lower two blocks to \( f_{a_k} \).

c. In the block for computing \( f_{a_k} \), replace “(A-1)” by “(B-5), in which \( x \equiv x \).”

d. In the block for computing \( f_{a_k} \), replace “(A-2)” by “(B-6), in which \( x \equiv x \).”

8) In Section V, change \( f_{a_k} \) in (31) and (32) to \( f_{a_k} \).

C. An Example

In order to compute \( \partial f_{a_k} / \partial \theta_{k,m} \) and \( \partial f_{a_k} / \partial \theta_{k,m} \) using (29) and (30), we need \( \partial \mu_{a_k} \) and \( \partial \nu_{a_k} \) and \( \partial \mu_{a_k} \) and \( \partial \nu_{a_k} \). Here, we will compute \( \partial \mu_{a_k} \) and \( \partial \nu_{a_k} \) for antecedent Gaussian primary MFs with uncertain means and input measurement Gaussian primary MFs with uncertain standard deviations. Formulas for antecedent MFs and their upper and lower MFs are given in Tables VII and VIII. Detailed derivations of these results can be found for product t-norm, in [6, pp. 394–399]. The results for minimum t-norm are derived in exactly the same manner as the ones for product t-norm. We summarize the parameters that \( \mu_{a_k} \) and \( \nu_{a_k} \) depend upon, as a function of \( x_k \), in Table IX. Its results were obtained by examining Table VII.

<table>
<thead>
<tr>
<th>( \theta_{k,1} )</th>
<th>( \theta_{k,2} )</th>
<th>( \theta_{k,3} )</th>
<th>( \theta_{k,4} )</th>
<th>( \theta_{k,5} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( m_{k,1} )</td>
<td>( m_{k,2} )</td>
<td>( m_{k,3} )</td>
<td>( m_{k,4} )</td>
<td>( m_{k,5} )</td>
</tr>
<tr>
<td>( \mu_{a_k} )</td>
<td>( \mu_{a_k} )</td>
<td>( \mu_{a_k} )</td>
<td>( \mu_{a_k} )</td>
<td>( \mu_{a_k} )</td>
</tr>
<tr>
<td>( \nu_{a_k} )</td>
<td>( \nu_{a_k} )</td>
<td>( \nu_{a_k} )</td>
<td>( \nu_{a_k} )</td>
<td>( \nu_{a_k} )</td>
</tr>
</tbody>
</table>

Results for \( \mu_{a_k} \) and \( \nu_{a_k} \) are given in Tables VII and VIII. Detailed derivations of these results can be found for product t-norm, in [6, pp. 394–399]. The results for minimum t-norm are derived in exactly the same manner as the ones for product t-norm. We summarize the parameters that \( \mu_{a_k} \) and \( \nu_{a_k} \) depend upon, as a function of \( x_k \), in Table IX. Its results were obtained by examining Table VII.

<table>
<thead>
<tr>
<th>( \mu_{a_k} )</th>
<th>( \nu_{a_k} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \mu_{a_k} )</td>
<td>( \nu_{a_k} )</td>
</tr>
</tbody>
</table>

You can compute the exact derivatives by using the formulas in Table VII.

The results in Tables VII and VIII were used in Figs. 2 and 4 as follows. The tests on variable \( x_k \) in VIII let us implement the top block of Fig. 4 in which we have to determine the active states. The results in Table X provide \( \partial \mu_{a_k} \) and \( \partial \nu_{a_k} \), which is needed to compute \( \partial f_{a_k} / \partial \theta_{k,m} \) and \( \partial f_{a_k} / \partial \theta_{k,m} \), and the results in Table V provide \( \partial \mu_{a_k} \) and \( \partial \nu_{a_k} \), which is needed to compute \( \partial f_{a_k} / \partial \theta_{k,m} \).
REFERENCES


Jerry Mendel (S’59–M’61–SM’72–F’83–LF’04) received the Ph.D. degree in electrical engineering from the Polytechnic Institute of Brooklyn, Brooklyn, NY.

Currently, he is Professor of Electrical Engineering and Associate Director for Education and Outreach of the Integrated Media Systems Center at the University of Southern California, Los Angeles, where he has been since 1974. He has published over 430 technical papers and is the author and/or editor of eight books, including Uncertain Rule-based Fuzzy Logic Systems: Introduction and New Directions (Upper Saddle River, NJ: Prentice-Hall, 2001). His present research interests include: type-2 fuzzy logic systems and their applications to a wide range of problems, including target classification and computing with words, and spatio-temporal fusion of decisions.

Dr. Mendel is a Distinguished Member of the IEEE Control Systems Society. He was President of the IEEE Control Systems Society in 1986, and is presently Chairman of the Fuzzy Technical Committee and a Member of the Administrative Committee of the IEEE Neural Networks Society. Among his awards are the 1983 Best Transactions Paper Award of the IEEE Geoscience and Remote Sensing Society, the 1992 Signal Processing Society Paper Award, the 2002 Transactions on Fuzzy Systems Outstanding Paper Award, a 1984 IEEE Centennial Medal, and an IEEE Third Millennium Medal.