Proceedings of the First International Conference on Mobile Computing Wireless Communication, E-Health, M-Health and Telemedicine (MWEMTeM 2008) ISBN: 978-2902-43-8

A FUZZY LOGIC TEMPERATURE CONTROLLER FOR PRETERM NEONATE INCUBATOR

[a]Bajeh, O. A. and [b]Emuoyibofarhe O. J.
[a]Department of Computer Science, Faculty of Communication & Information Science, University of Ilorin, Ilorin, Nigeria.
Email: bajeh_amos@yahoo.com or bajehamos@unilorin.edu.ng.

[b]Department of Computer Science, Ladoke Akintola University, Ogbomoso, Nigeria.
Email: eojustice@justice.com or eojustice@gmail.com

ABSTRACT

Preterm Neonates come into the world earlier than their expected period/time with premature body system and organ(s). They lack the sufficient body fat necessary to generate heat and maintain their body temperature, thus one of the fundamental requirements for the survival of these Neonates is ***thermoneutrality*** *- An environment temperature at which minimal rate of oxygen requirement or energy expenditure occurs. Incubators and Radiant warmers under the close watch of expert(s) are used to keep the Neonates warm to avoid a state of* ***hypothermia*** *or* ***hyperthermia****. This is error-prone and most often inefficient in the care for the Neonates.*

This research work employs the use of Fuzzy Logic Control (FLC) Methodology to control the supply of heat into the Neonate Incubator in order to attain thermoneutrality. Fuzzy Logic Controllers have proven to be as efficient as (if not better than) Conventional Controllers and sometimes the only alternative suitable for designing controllers for complex processes/plants or for processes that are non-Linear in nature. In this research a ***2 inputs, 2 outputs, 18 MISO Rules PD Fuzzy Logic Temperature Controller*** *was designed and implemented.*

This research work did not only show the applicability of Fuzzy Logic Control Methodology to controlling the Incubator Temperature to attain thermoneutrality but also the efficient stabilization of the Incubator Temperature at a desired value and thereby prevention of hypothermia/hyperthermia related diseases/conditions and death. Also it eliminates the error-prone human control of the heating element of the Neonate Incubator.

The performance evaluation of the designed Fuzzy Logic Controller was conducted through the examination of the stability, rise time, over/under shoot of the controller in a Computer Simulation using C++ programming Language. The Controller's reasoning procedure was examined using the plotting capabilities provided in MatLab 7.0. The results show that the Fuzzy Logic Controller provides a desirable performance in the attainment of thermoneutrality for preterm neonates.

Keywords: *Fuzzy Logic, Fuzzy Controller, Membership Function (MF), Incubator, Neonates, Temperature.*

1.0 INTRODUCTION.

A Control System is a group of components assembled in such a way as to regulate energy input to achieve a desired output. The function of a Control System is to maintain certain essential properties of a system at a desired value.

Preterm Neonates (also known as premature Infants) are babies born earlier than the time medically expected for delivery. This occurs when the gestation period of 38 – 40 weeks is not attained. Preterm Neonates cannot produce heat and regulate their body temperature as a result of poor muscular development, inability to use their sweat glands, poor development of the brown adipose tissue for thermogenesis – a non-shivering process of generating heat. These Neonates, been a special Infants, require special attention and care; thus, the use of the Incubators/Radiant warmers to regulate the temperature and also humidity and ventilation of the Neonates. An Incubator comprises a transparent chamber and the components that regulate its temperature, humidity and ventilation. It usually has openings for ventilation and via which the Neonate can be accessed and taken care of when need be.

Fuzzy Controller employs Fuzzy Logic in the quantification of linguistic descriptions and in carrying out reasoning which are approximate in nature. Fuzzy Logic is the logic of fuzzy set which is a set without a crisp boundary but where the transition from "belong to a set" to "not belong to a set" is gradual and not sharp as in classical set [1]. The smooth transition in fuzzy set is characterized by the use of Membership Functions that gives fuzzy set flexibility in modeling commonly used linguistic expressions or terms such as "the water is hot" or "the temperature is high" [1]. Fuzzy Control, thus, provides a formal methodology for representing, manipulating and implementing a human heuristic knowledge about how to control a system [2]. Some systems complexity gets high so that the control of such system can only be achieved through the use of human heuristic knowledge (fuzzy) rather than mathematical models (conventional control).

2.0 RELATED WORK

Fuzzy Logic has been employed as a methodology for the design of Control Systems. It has served as a good (if not better) alternative methodology for Controller designs. Some of the related works identified in the course of this research work are:

- The Reactor Temperature Controller as reported in [26], where a product of Aptronix called FIDE was used to design a fuzzy Controller for the control of the temperature in a Reactor.
- Temperature Control: PID vs Fuzzy Logic as reported in [9] where a Fuzzy Logic version of a PID Home Temperature Control System was reported designed and its advantage over the conventional PID system was highlighted.
- Balancing of an inverted pendulum in a vertical position on a Cart using Fuzzy Control as reported in [2] where a PD Fuzzy Control system was designed. Its performance evaluation was reported done using a simulation of the control system.
- Others as reported in [2] are: Vibration Damping for a Flexible-Link Robot, Rotationally inverted Pendulum, Machine Scheduling and A Fault Detection System for Aircraft. Also reported in [6] is a Fuzzy logic Speed Control which can also be referred to as Cruise Control.

3.0 DEFINITION OF TERMS

Suppose X is a space of objects and x is a generic element of $X(x \in X)$; a classical conventional set $A(A \subseteq X)$ is defined as a collection of elements or objects $x \in X$ such that each x can either belong or not belong to the set A. This can be mathematically defined by defining a function for each element x in X. This function is referred to as **characteristic function**. The set A can be represented by a set of order pairs: $(x,0)$ or $(x,1)$ which indicates $x \notin A$ or $x \in A$ respectively [1]
Unlike the aforementioned classical set, a fuzzy set expresses the degree to which an element belongs to a set and this degree is represented by a function which is allowed to have values between 0 and 1.0. This function is called The **Membership Function (MF)** and its values denote the degree of membership of the element(s) in a given set.

3.1 Terminology

Definition1: Fuzzy sets and MFs

If X is a collection of objects denoted generically by x, then a fuzzy set A on X is defined as a set of ordered pairs. $A = \{(x, \mu_A(x)) / x \in X \}$

where $\mu_A(x)$ is called the membership function (MF) for the fuzzy set A. The MF maps each element of X to a membership grade/value in a range $[0,1]$ [1]. This MF describes the "certainty" that an element x is the member of the set A. X is referred to as the **Universe of Discourse** or simply **Universe** which is a way to say all the objects in the universe of a particular kind [6]. It may consist of discrete objects (ordered or non-ordered) or continuous space.

Definition 2: Fuzzy Subset [2]

Given two fuzzy set A and B on the same universe of discourse X with membership function denoted as $\mu_A(x)$ and $\mu_B(x)$. A is defined to be a "Fuzzy Subset" of B denoted as $A \subset B$ if $\mu_A(x) \le \mu_B(x) \quad \forall x \in X$. This is depicted in figure 3.1 (a).

Definition 3: Fuzzy Support [1]

The support of a fuzzy set A is the set of the points x in X such that $\mu_A(x) > 0$ i.e.

$Support(A) = \{x / \mu_A(x) > 0\}$. This is depicted in figure 3.1 (b).

Definition 4: Fuzzy Core [1]

The Core of a fuzzy set A in the set of all points x in X such that $\mu_A(x) = 1$ i.e.

$Core(A) = \{x / \mu_A(x) = 1\}$. This is depicted in figure 3.1 (b).

Definition 5: Crossover Points

A Crossover point of a fuzzy set A is a point $x \in X$ at which $\mu_A(x) = 0.5$

$CrossOver(A) = \{x / \mu_A(x) = 0.5\}$. This is depicted in figure 3.1 (b)

Definition 6: Fuzzy Singleton

Fuzzy set whose support is a single point in X with $\mu_A(x) = 1$ is called a fuzzy singleton

Definition 7: Open Left, Open Right, Closed MFs

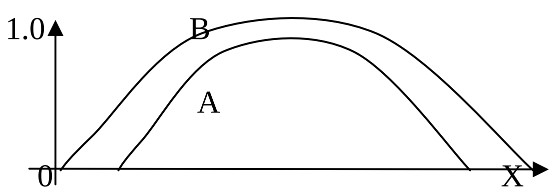
A fuzzy set A is **Open left** if $\lim_{x \to -\infty} \mu_A(x) = 1$ and $\lim_{x \to +\infty} \mu_A(x) = 0$

Open right if $\lim_{x \to -\infty} \mu_A(x) = 0$ and $\lim_{x \to -\infty} \mu_A(x) = 1$

Closed if $\lim_{x \to -\infty} \mu_A(x) = \lim_{x \to -\infty} \mu_A(x) = 0$

These are depicted in figure 3.2

(a) A⊂B



(b)

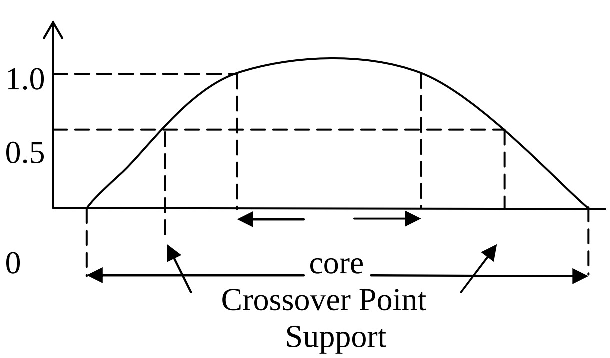


Figure 3.1

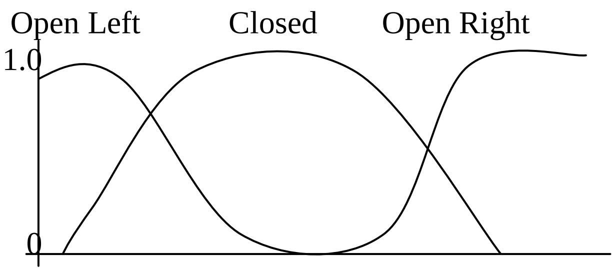


Figure 3.2: Open Left, Open Right, and Closed MF.

3.2 Set Theoretic Operations

Definition 8: Fuzzy set Equality [5]

The fuzzy sets A and B are said to be equal if their MF are equal at every point in the universe of discourse, i.e. $A = B$ if $\mu_A(x) = \mu_B(x) \quad \forall x \in X$.

Definition 9: Fuzzy Complement (Negation) [2]

The complement ("not") of a fuzzy set A with the MF $\mu_A(x)$ is noted by $\mu_{\bar{A}}(x)$ and has a MF given by $1-\mu_A(x)$, i.e. $\mu_{\bar{A}}(x) = 1-\mu_A(x)$.

Definition 10: Fuzzy Set Union (OR)

The union of two fuzzy sets A and B defined over the same universe of discourse X is a new fuzzy set denoted as $A \cup B$ with a MF that represent the maximum degree of relevance between each element and the new fuzzy. [5]. The class of fuzzy union operators use in fuzzy inferences are referred to as $T-Conorm$ (or $S-norm$) operators. The following are four $T-Conorm$ operators commonly used.

(a) $\max imum : S(a,b) = \max(a,b) = a \vee b$

(b) $Algebraic \quad sum : S(a,b) = a + b - ab$

(c) $Bounded \quad sum : S(a,b) = 1 \wedge (a+b)$

(d) $Drastic \quad sum : S(a,b) = \begin{cases} a & if \quad b = 0 \\ b & if \quad a = 0 \\ 1 & if \quad a,b > 0 \end{cases}$

Definition 11: Fuzzy Set Intersection (AND)

The intersection of two fuzzy sets A and B defined over the same Universe of discourse X, is a new fuzzy set denoted as $A \cap B$ with a MF that represent the minimum degree of relevance between each element and the new fuzzy set [5]. The class of fuzzy intersection operators used in fuzzy inferences is referred to as $T-norm$ (Triangular norm). The following are four of the most frequently used $T-norm$ operators

(a) minimum: $T_{\min}(a,b) = \min(a,b) = a \wedge b$

(b) $Algebraic \ product : T_{ap}(a,b) = ab$

(c) $Bounded \quad product : T_{bp}(a,b) = 0 \vee (a+b-1)$

(d)

$$\text{Drastic product} : T_{dp}(a,b) = \begin{cases} a & \text{if} \quad b = 1 \\ b & \text{if} \quad a = 1 \\ 0 & \text{if} \quad a,b > 1 \end{cases}$$

Crisp Input(s)

3.3 One Dimensional MF formulation and Parameterization

Fuzzy sets are completely characterized by their membership functions (MF). In practice the universe of discourse X is a real line R, thus it is impracticable to list all the pairs $(x, \mu_A(x))$ defining the membership function of a fuzzy set. A more concise and convenient way to define a MF is to express it as a mathematical formula [1]. Below is the most commonly used MF formulation among others:

Triangular MF [1]: A triangular MF is specified by three parameters $\{a, b, c\}$ as follows:

$$\text{triangle}(x : a, b, c) = \begin{cases} 0 & x \le a \\ \dfrac{x-a}{b-a} & a \le x \le b \\ \dfrac{c-x}{c-b} & b \le x \le c \\ 0 & c \le x \end{cases}$$

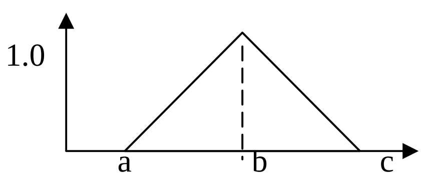


Figure 3.3 Triangular MF

Other MF that have been used in the design of Fuzzy Controllers are **Gaussian MF, Trapezoidal MF , Generalized bell MF** and **Sigmoidal MF.**

3.4. Fuzzy Controller: Generally, a fuzzy system is a static non-linear mapping between its input(s) and output(s). A Fuzzy System has inputs $u_i, i = 1,2,.....n$, outputs v_j, $j = 1,2,......m$ and other components as shown in figure 3.4 below [2]. The inputs and outputs are 'crisp' (i.e. real number). The **fuzzification** block converts the crisp inputs to fuzzy sets, the **inference mechanism** uses the fuzzy rules in the **rule-base** to produce fuzzy conclusions (implied fuzzy sets) and the **defuzzification** block converts these fuzzy conclusions into the crisp outputs.

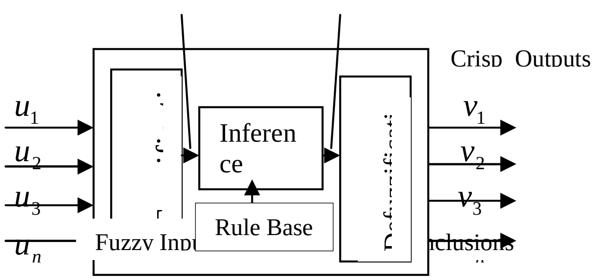


Figure 3.4 Fuzzy Controller

4.0 THE DESIGN

4.1 The Controller's Inputs:

1. **Error in Temperature**: The difference between the desired value (also known as the reference temperature) and the actual Incubator temperature at a particular time.
2. **Change in error in Temperature**: Additional information of whether the temperature is increasing, decreasing or stable will further enhance the ability of the controller to control the temperature. This is

change in error and it is derived by taking the derivative of the error is temperature with respect to time

4.2 The Controller's Outputs:

1. **Voltage**: Electrically the voltage across the heating element can determine the quantity of heat generated by the heater. This implies that controlling the voltage across the heater will determine the quantity of heat supply to the Incubator and thereby controlling the temperature
2. **Pressure**: The removal of heat when the temperature goes beyond the reference point is not done by voltage change. High pressure results in high temperature, and low pressure results in low temperature [20]. Taking the advantage of the influence of pressure on temperature, the temperature of the Incubator can be reduced by reducing the pressure in the chamber. This reduces the kinetic energy of the air and allows the flow of fresh and cooler air into the chamber from the surrounding.

Having defined the controller's inputs and outputs the control system can be specified as follows:

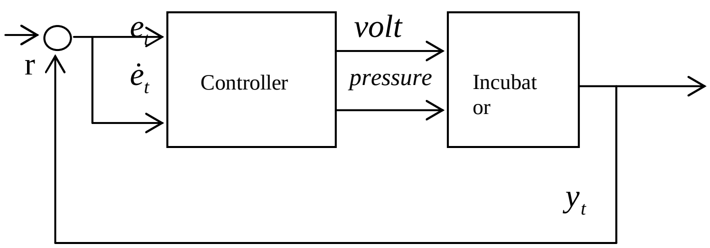
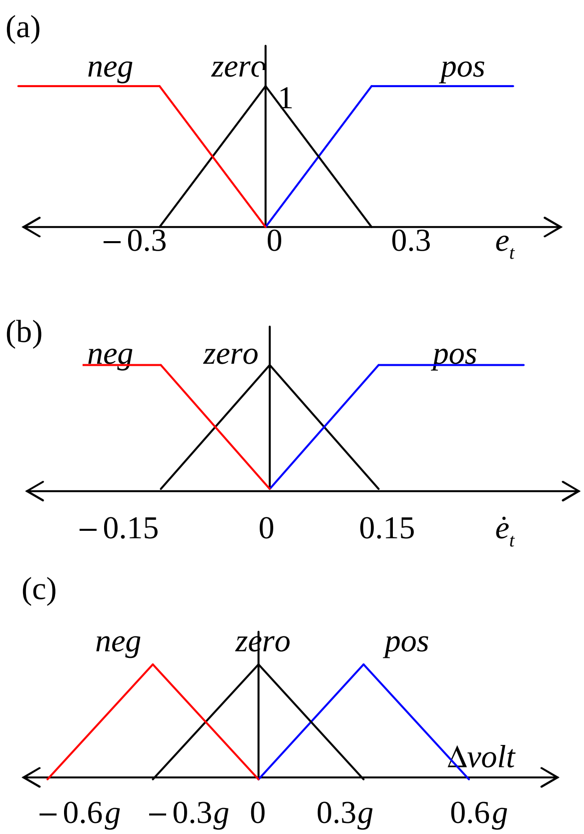


Figure 4.1 fuzzy Temperature Controller

4.3 Linguistic Variables, Linguistic Value, MFs and Universe Of Discourse

The controller's inputs and outputs are the systems linguistic variables and they are error in temperature (e_t), change in error ($\dot{e}_t$) change in voltage ($\Delta volt$) and pressure (p_0). There designs are graphically shown in figure 4.2.



(d)

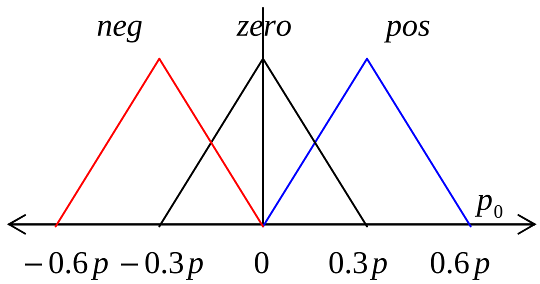


Figure 4.2. Linguistic Variables and MFs

4.4. The Controller's Linguistic Rules (Fuzzy If Then Rules)

Based on the choice of inputs and outputs as discussed in 4.1 and 4.2, the following fuzzy **IF THEN Rules** are specified as a set of rules that represents an expert knowledge about how to best control the Incubator temperature.

Rule 1: **IF** e_t is neg **AND** $\dot{e}_t$ is neg **THEN** $\Delta volt$ is neg **AND** p_0 is neg

Rule 2: **IF** e_t is neg **AND** $\dot{e}_t$ is $zero$ **THEN** $\Delta volt$ is neg **AND** p_0 is neg

Rule 3: **IF** e_t is neg **AND** $\dot{e}_t$ is pos **THEN** $\Delta volt$ is $zero$ **AND** p_0 is $zero$

Rule 4: **IF** e_t is $zero$ **AND** $\dot{e}_t$ is neg **THEN** $\Delta volt$ is neg **AND** p_0 is $zero$

Rule 5: **IF** e_t is $zero$ **AND** $\dot{e}_t$ is $zero$ **THEN** $\Delta volt$ is $zero$ **AND** p_0 is $zero$

Rule 6: **IF** e_t is $zero$ **AND** $\dot{e}_t$ is pos **THEN** $\Delta volt$ is pos **AND** p_0 is $zero$

Rule 7: **IF** e_t is pos **AND** $\dot{e}_t$ is neg **THEN** $\Delta volt$ is $zero$ **AND** p_0 is $zero$

Rule 8: **IF** e_t is pos **AND** $\dot{e}_t$ is $zero$ **THEN** $\Delta volt$ is pos **AND** p_0 is $zero$

Rule 9: **IF** e_t is pos **AND** $\dot{e}_t$ is pos **THEN** $\Delta volt$ is pos **AND** p_0 is $zero$

The rules are MIMO rules which can be represented in the MISO form [2] as follows

Rule 1: **IF** e_t is neg **AND** $\dot{e}_t$ is neg **THEN** $\Delta volt$ is neg

Rule 2: **IF** e_t is neg **AND** $\dot{e}_t$ is neg **THEN** p_0 is neg

Rule 3: **IF** e_t is neg **AND** $\dot{e}_t$ is $zero$ **THEN** $\Delta volt$ is neg

Rule 4: **IF** e_t is neg **AND** $\dot{e}_t$ is $zero$ **THEN** p_0 is neg

Rule 5: **IF** e_t is neg **AND** $\dot{e}_t$ is pos **THEN** $\Delta volt$ is $zero$

Rule 6: **IF** e_t is neg **AND** $\dot{e}_t$ is pos **THEN** p_0 is $zero$

Rule 7: **IF** e_t is $zero$ **AND** $\dot{e}_t$ is neg **THEN** $\Delta volt$ is neg

Rule 8: **IF** e_t is $zero$ **AND** $\dot{e}_t$ is neg **THEN** p_0 is $zero$

Rule 9: **IF** e_t is $zero$ **AND** $\dot{e}_t$ is $zero$ **THEN** $\Delta volt$ is $zero$

Rule 10: **IF** e_t is $zero$ **AND** $\dot{e}_t$ is $zero$ **THEN** p_0 is $zero$

Rule 11: **IF** e_t is $zero$ **AND** $\dot{e}_t$ is pos **THEN** $\Delta volt$ is pos

Rule 12: **IF** e_t is $zero$ **AND** $\dot{e}_t$ is pos **THEN** p_0 is $zero$

Rule 13: **IF** e_t is pos **AND** $\dot{e}_t$ is neg **THEN** $\Delta volt$ is $zero$

Rule 14: **IF** e_t is pos **AND** $\dot{e}_t$ is neg **THEN** p_0 is $zero$

Rule 15:**IF** e_t is pos **AND** $\dot{e}_t$ is $zero$ **THEN** $\Delta volt$ is pos

Rule 16:**IF** e_t is pos **AND** $\dot{e}_t$ is $zero$ **THEN** p_0 is $zero$

Rule 17:**IF** e_t is pos **AND** $\dot{e}_t$ is pos **THEN** $\Delta volt$ is pos

Rule 18:**IF** e_t is pos **AND** $\dot{e}_t$ is pos **THEN** p_0 is $zero$

4.5 The Controller's Fuzzification Interface

The singleton fuzzification is used in the controller. This fuzzification is widely use due to its computational simplicity. Given an input u_1 (for e_t) and u_2 (for $\dot{e}_t$) into the controller, a representative fuzzy sets $A_{e_t}^{FUZ}$ and $A_{\dot{e}_t}^{FUZ}$ with their respective MFs defined below will be derived.

$$\mu_{A_{e_t}^{FUZ}}(x) = \begin{cases} 1 & x = u_1 \\ 0 & otherwise \end{cases} \quad \text{and} \quad \mu_{A_{\dot{e}_t}^{FUZ}}(x) = \begin{cases} 1 & x = u_2 \\ 0 & otherwise \end{cases}$$

4.6 The Controller's Inference Mechanism

The inference mechanism has 2 basic tasks:

Task1: Matching: The fuzzification produces fuzzy sets which the inference mechanism takes as input. The matching takes the following two steps

Step 1: Combine controller input with rule premises: This is the scaling of the inputs fuzzy sets by the singleton fuzzy set input. It simply reduces to computing the membership value of the input fuzzy sets for the given inputs x_1 and x_2.

Step 2: Determine which rules are on: This step determines the membership values μ_i for the *ith* rule's premise (i.e. a value for $\mu_{premise}$ for the rule is determined) and it represents the certainty that each rule premise holds for the given input. This is defined as $\mu_{premise} = \mu_i(x_1, x_2) = \mu_{A_{e_t}^i}(x_1) * \mu_{A_{\dot{e}_t}^j}(x_2)$. Rules with $\mu_i > 0$ are considered to be on or active or fired at the state of the inputs x_1 and x_2. The operator $*$ is chosen to be the ***min*** operator.

Task 2: Inference Step: This task computes for the *ith* rule an implied fuzzy set, *B* with MF $\mu_{B_i}(y) = \mu_i * \mu_{B^i}(y)$. The implied fuzzy set B_i is a consequent fuzzy set that specifies the certainty level that the output should be a specific crisp output *y* on the universe of discourse of the output, when only the *ith* rule is taken into consideration. In a similar manner the implied fuzzy set B_i for all the rules that are on are computed.

4.7 The Controller's Defuzzification Interface.

The commonly used defuzzification method called the Centre of Gravity (COG) is used in this research to compute the crisp value for each of the controller's outputs. The COG defuzzification is determined as:

$$U^{Crisp} = \frac{\sum_{i=1}^{R} b_i^q \int_{y_q} \mu_{B_q^i}(y_q) dy_q}{\sum_{i=1}^{R} \int_{y_q} \mu_{B_q^i}(y_q) dy_q} \qquad 4.1.7$$

where *R* in the number of rule, b_i^q is the centre of the area of the membership function B_q^p associated with the implied fuzzy set B_q^i of the *ith* rule.

4.8 Incubator Temperature Model

The Neonate Incubator is an enclosed chamber made up of a transparent material with apertures for ventilation. Within the chamber is the heating element which provides heat into the chamber. The voltage output of the temperature controller is set across this heating element to generate heat which is passed into the chamber. The electrical heat that will be generated by the element is given by $Q = IVt$ (4.2.1)

For the air in the Incubator, the temperature change when Q quantity of heat is passed into the chamber is given by

$$Q = mc\theta \qquad (4.2.2)$$

where θ is change in temperature, m is mass of air in the chamber and c is the specific heat capacity of air. Let T_1 be the initial temperature and T_2 be the final temperature of the air after Q quantity of heat is supplied, then (4.2.2) becomes

$$Q = mc(T_2 - T_1) \qquad (4.2.3)$$

Equating (4.2.1) and (4.2.3),

$$IVt = mc(T_2 - T_1) \qquad (4.2.4)$$

Heat will be lost from the chamber to the surrounding via the apertures. Let this quantity of heat lost be h_L. (4.2.4) becomes

$$IVt = mc(T_2 - T_1) + h_L \qquad (4.2.5)$$

$$T_2 = T_1 + \frac{IVt - h_L}{mc} \qquad (4.2.6)$$

Controlling the voltage across the element controls the quantity of heat generated by the element and passed into the chamber. It does not reduces the temperature if need be. Thus, to implement the reduction of temperature, a piston can be used to control the pressure in the chamber. Whenever the piston is pulled the volume of the Incubator increases and so cooler air from the surrounding environment flows into the chamber thereby reducing the temperature. The volume of inflowing air will be equal to the change in volume of the chamber due to the pull on the piston.

Let T_i be the temperature of the inflowing air (i.e. the temperature of the surrounding environment), then Total heat supplied = heat lost by enclosed air + heat gained by cooler air

$$IVt = mc(T_1 - T_2) + m_i c(T_2 - T_i)$$

$$\Rightarrow T_2(m_i c - mc) = IVt - mcT_1 + m_i cT_i \; T_2 = \frac{IVt - mcT_1 + m_i cT_i}{(m_i c - mc)} \qquad (4.2.7)$$

Equation (4.2.7) models the temperature change when the piston is pulled and cooler air flows into the chamber. Equations (4.2.6) and (4.2.7) model the temperature dynamics of the Neonate Incubator in terms of voltage change and pressure change (implemented by using a piston).

5.0 IMPLEMENTATION

5.1 The Fuzzy Controller Implementation.

The Fuzzy Controller was programmed in the C++ programming language. This provides some advantage by making it easier to transfer the code directly to an experimental setting for real-time control. MatLab 7.0 was also used to program the controller because it provides a further test for the designed controller and provides plotting capabilities from which the plotted graph for the rule evaluation and surface graph of the controller are derived. The computer program pseudocode use to develop the controller is stated below.

Fuzzy Controller pseudocode

1. Obtain e and $\dot{e}$ values

   (Gets inputs to the fuzzy controller)

2. Compute $mf1[i]$ and $mf2[j]$ for all i, j

   (find the values of all MFs given the values of e and $\dot{e}$)

3. Compute $prem[i][j] = \min(mf1[i], mf2[j])$ for all i, j

(Find the value for the premise MF for a given e and $\dot{e}$ using the **min** operation)

4. Compute for all i, j

$impVolt[i][j] = areaimp(ruleVolt[i][j], prem[i][j])$ and

$imp\Pr essure[i][j] = areaimp(rule\Pr essure[i][j], prem[i][j])$ (Find the area under the MFs for possible implied Fuzzy set for the two inputs)

5. Let $num = 0$, $denom = 0$, $num2 = 0$ and $denom2 = 0$

6. For $i = 0$ to 2,

   For $j = 0$ to 2

   { $num = num + impVolt[i][j] * centre(ruleVolt[i][j])$ (Compute numerator for COG for the Volt)

$denom = denom + impVolt[i][j]$

(Compute denominator for COG for the Volt)

$num2 = num2 + imp\Pr essure[i][j] * centre(rule\Pr essure[i][j])$ (Compute numerator for COG for the Pressure)

$denom2 = denom2 + imp\Pr essure[i][j]$ (Compute denominator for COG for the Pressure)

   }

7. $\Delta volt = num / denom$

   (Compute change in voltage)

   $\Delta pressure = num2 / denom2$

   (Compute change in pressure)

8. Output $voltage = voltage + \Delta volt$ and $pressure = pressure + \Delta pressure$

9. Go to step 1.

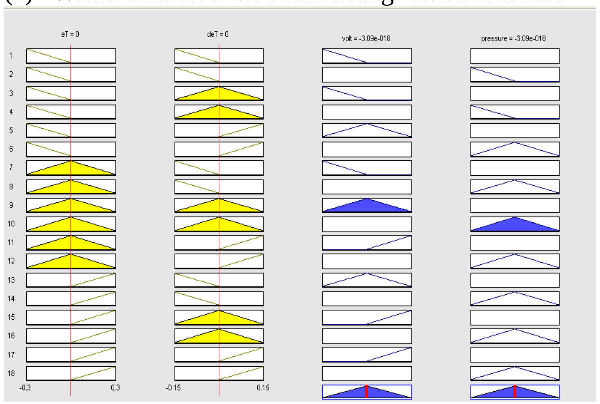
6.0 EVALUATION

6.1 Rule Evaluation

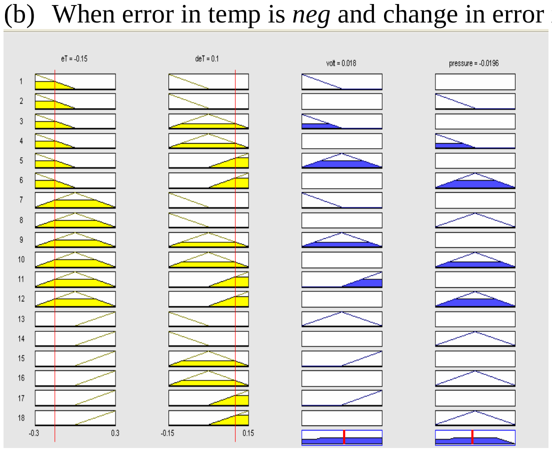
The decision making procedure of the Fuzzy Controller can be depicted (as shown below) by evaluating the rules at an input instant i.e. given a set of values for the inputs e and $\dot{e}$ the steps that will be taken by the controller to arrive at its outputs can be graphically shown. The rule evaluation for some instant of e

and $\dot{e}$ is shown in figures 6.1 (a) and (b). Figures 6.1 (c) and (d) shows the Surface for the Voltage and Pressure change of the Controller.

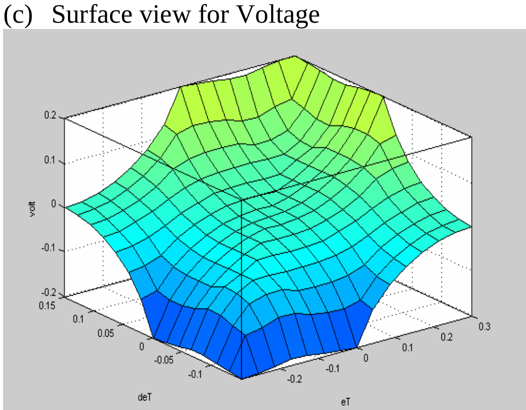
(a) When error in is *zero* and change in error is *zero*



(b) When error in temp is *neg* and change in error is *pos*



(c) Surface view for Voltage



(d) Surface view for Pressure

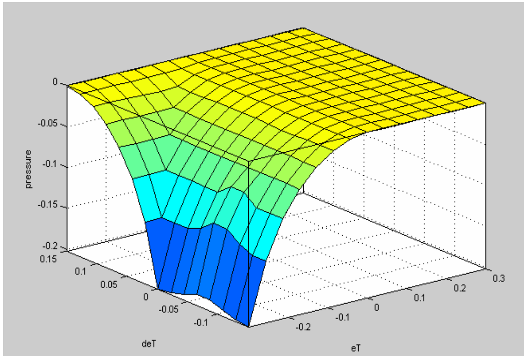
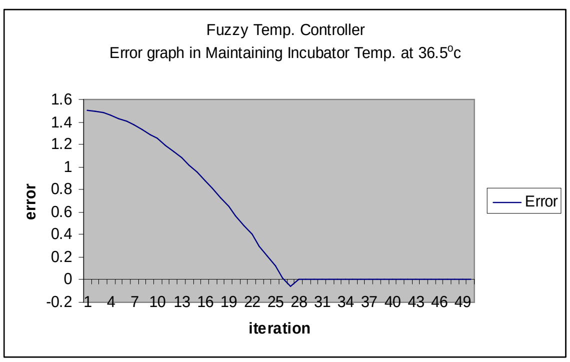


Figure 6.1 Rule Evaluation

6.2 Performance Evaluation

Having developed the computer program to simulate the designed Fuzzy Controller, the model developed for the temperature dynamics of the Incubator was implemented in C++ programming language. These two programs were harmonized to represent a simulation for the entire control system. The data output from the simulation program were used to plot the graphs in figures 6.2, 6.3 and 6.4. These graphs depict the dynamics of the system which shows how the Incubator temperature is maintained by varying the voltage across the heating element and moving the piston as to stabilize the temperature within the desired point. The graphs also show the effect of varying the output **scaling gain** on the performance of the Controller.

(a)



(b)

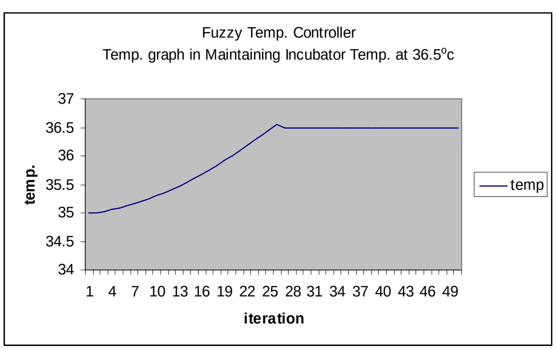
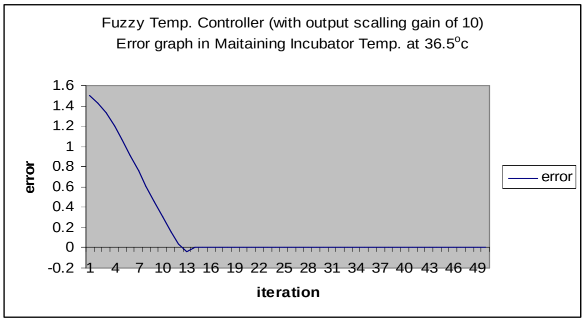


Figure 6.2. The Fuzzy Controller Performance

(a)



(b)

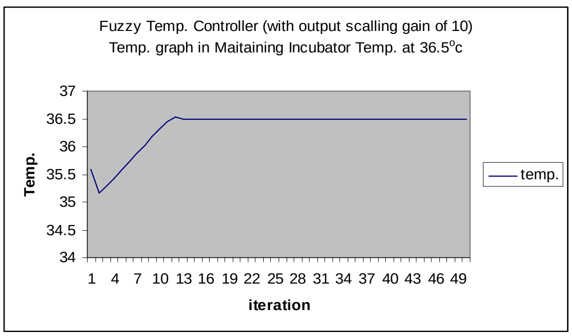
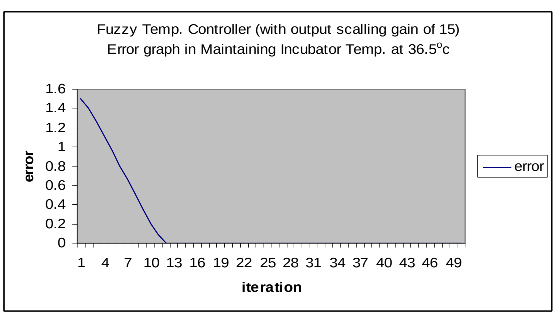


Figure 6.3. The Fuzzy Controller Performance (with output scaling gain of 10)

(a)



(b)

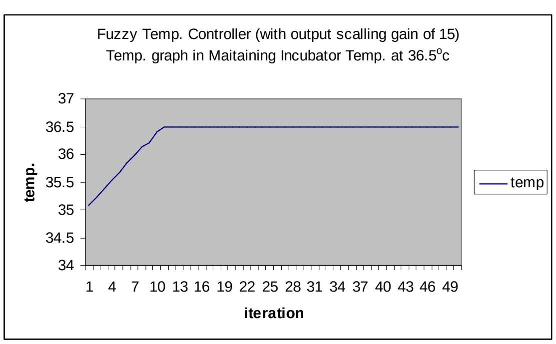


Figure 6.4. The Fuzzy Controller Performance (with output scaling gain of 15)

7.0 DISCUSSION OF RESULT

The performance evaluation conducted in section 6.0 from which the graphs in figures 6.2, 6.3 and 6.4 are derived has shown that the Controller designed meets the closed-loop specification of maintaining the Incubator Temperature at a desired value with allowable rise time, over/under shoot and stability. Figure 6.2 (a) shows how the error in Temperature falls until it becomes zero, thereby indicating a NORMAL Temperature i.e. a Temperature at the desired point/value. Figure 6.2 (b) is the corresponding graph showing how the temperature rises to NORMAL as the error falls to zero. This graphs shows that the Temperature becomes NORMAL at the 25th iteration of the close-loop system. Figure 6.3 further shows an enhanced version of the Controller. The use of Scaling Gain to boost the Voltage output of the Controller reduced the number of iteration to 12 thereby making it faster in attaining the desired Temperature. In this figure a Scaling gain of 10 was used. In trying to find an optimum value for the Scaling gain, 15 and 20 were further tried as Scaling gain; though the improvement observed was just by an iteration, that of 20 showed a wavy nature where at certain intervals the Incubator Temperature overshoots and then stabilizes but that of 15, depicted in figure 6.4, shows a stable state after attaining the NORMAL Temperature with little or no overshoot. Thus 15 is the suitable scaling gain for the Controller.

7.1 Contribution to Knowledge

This research work has shown the applicability of Fuzzy Logic methodology in controlling the temperature of a preterm neonate. The enhancement of the performance of the Controller by the use of output scaling gain improves the rise time of the controller thereby showing that the Controller will be able to reduce the effect of any disturbance on the temperature regulation within a very short time. Also the measure of the overshoot falls between 0 and 0.1 oc which is very well between the allowable change of only 0.2 – 0.3 ^{o}c. These are the contributions to knowledge that this research work has achieved.

7.2 Further Research Work and Conclusion

Further research works that will be done are:

- Design the controller using other parameterized one-dimensional MFs and conduct a comparative analysis of the effect on the performance of the controller.
- Increase the number of MFs in the design and study the effects this will have on the performance of the Controller.
- Modify the design of the Controller for a Transport Incubator.

In conclusion, this research work has further shown the applicability of Fuzzy Logic Control Methodology in achieving a good and stable control of a System.

8.0 REFERENCES

[1] J.S.R. Jang, C.T. Sun and E. Mizutani (1997); **Neuro-fuzzy and Soft Computing** (A Computational Approach to Learning and Machine Intelligence); Prentice Hall, Inc.; pp 1- 90.

[2] Kevin M. Passino & Stephen Yurkovich (1998); **Fuzzy Control;** Addison Wesley Longman Inc. Carlifornia

[3] Mupanemunda R.H. & Watkinso M. (1999), Key Topics in Neonatology Bios Scientific Publishers, Oxford.

[4] http://www/madehow.com/

[5] N.P. Padhy (2006); **Artificial Intelligence and Intelligent systems**; Oxford University Press.

[6] http://www.fuzzy-logic.com; **Fuzzy logic Tutorial**

[7] M.N. Cirstea, A. Dinu, J.G. Khor, M. McCormink (2002); **Neural and Fuzzy Logic Control of Daves and Power.**

[8] NNF Teaching Aids: **Newborn Care. Hypothermia in Newborn**

[9] http://www.controleng.com/; **Temperature control: PID vs. Fuzzy Logic**

[10]Tom Duncan (1982): Physics A textbook for Advance Level Students; John Murray Publishers Ltd, London; pp 36-82, 427-456

[11]http://web.uci.ac.za/;**Temperature Control.**

[12]WHO/RHT/MSM/97.2;**Thermal Protection of the Newborn –a practical guide**

[13] Adamsons K, Towel M. (1965); **Thermal Aomeostasis in the fetus and Newborn;** Anesthesiology, 26 531 – 548.

[14] Brick K. (1961); **Temperature regulation in the newborn Infant;** Bio Neonate 3, pp 65.

[15] Karlsson H.(1996); **Skin-to-skin care: Heat balance Arch Dis Child,** 75: F130 –

F132.

[16] Aiolet D et al. (1989); **Oxygenation, heat rate and temperature in very low birth weight Infants during skin-to-skin contact with their mothers;** Acta Pedscan 78, pp89 – 93.

[17] Dahn LS, James Ls. (1972); **Newborn temperature and calculated heat loss in the delivery room.** Pediatries 49, pp504 – 513.

[18] P. B. Shola (2008); **Data Structure And Algorithms using C++;** Reflect Publisers Nig.

[19] L. A. Zadeh (1965); **Fuzzy Sets**; Information & Control, vol 8, pp 338- 353.

[20] L. A. Zadeh (1983); **A Fuzzy-Set-Theoretic Approach to the Compositionality of Meaning: Propositions, Disposition and Canonical Forms**; Journal of Semantic, An Int'l Journal, Vol. 11, No. 3/4.

[21] L. A. Zadeh (1979); **Fuzzy Sets and Information Granularity;** Selected Papers by L.A. Zadeh; pp 443-448.

[22] L. A. Zadeh (1975); **The Concept of a Linguistic Variable and its Application to Approximate Reasoning-I**; Information Science 8, pp199-249.

[23] L. A. Zadeh (1975); **The Concept of a Linguistic Variable and its Application to Approximate reasoning-II**; Information Science 8, pp301-357.

[24] L. A. Zadeh (1975); **The Concept of a Linguistic Variable and its Application to Approximate reasoning-III**; Information Science 9, pp48-80.

[25] L. A. Zadeh; **A Computational Theory of Disposition.**

[26]http://www.aptronix.com/fuzzynet/applnote/reactor.htm

[27] Flenady VJ, Woodgate PG. (2001): **Radiant warmers versus incubators for regulating body temperature in newborn infants.** Cochrane Library, Issue 1. Oxford.

[28] B. Chike Obi (1989): **Introduction to Thermal Physics**; Heinemann Educational Books (Nig.) Ltd.