THE UNIVERSITY OF WESTERN ONTARIO DEPARTMENT OF CIVIL AND ENVIRONMENTAL ENGINEERING

Water Resources Research Report

Application of the Fuzzy Performance Indices to the City of London Water Supply System

> By: Ibrahim El-Baroudy and Slobodan P. Simonovic



Report No: 050 Date: January 2005

ISSN: (print) 1913-3200; (online) 1913-3219; ISBN: (print) 978-0-7714-2630-8; (online) 978-0-7714-2631-5;

APPLICATION OF THE FUZZY PERFORMANCE INDICES TO THE CITY OF LONDON WATER SUPPLY SYSTEM

BY

Ibrahim El-Baroudy, Ph.D. Candidate Department of Civil and Environmental Engineering University of Western Ontario, London, Ontario

AND

Slobodan S. Simonovic, Professor and Research Chair Department of Civil and Environmental Engineering Institute for Catastrophic Loss Reduction University of Western Ontario, London, Ontario

January, 2005

TABLE OF CONTENT

1 IN	FRODUCTION	1
1.1 (Dbjectives of the analysis	1
1.2 I	Report organization	1
1.3 8	Summary of the results	2
2 SY	STEM DESCRIPTION	3
2.1 I	ake Huron primary water supply system (LHPWSS)	3
2.1.1	Intake system	5
2.1.2	Water treatment system	7
2.1.3	Conveyance and storage systems	
2.2 H	Elgin area primary water supply system (EAPWSS)	9
2.2.1	Intake system	11
2.2.2	Water treatment system	11
2.2.3	Conveyance and storage systems	12
3 MB	CTHODOLOGY FOR SYSTEM RELAIBILITY ANALYSIS	13
3.1 N	Aulti-component system representation	13
3.2 (Capacity and requirement of system components	15
3.2.1	System component capacity membership function	19
3.2.2	System component requirement membership function	21
3.2.3	Standardization of membership functions	22
3.3 (Calculation of fuzzy performance indices	24
3.3.1	System-state membership function	25
3.3.2	Acceptable level of performance membership function	27

3.	3.3	System-failure membership function	29
3.	3.4	Fuzzy reliability-vulnerability index	32
3.	3.5	Fuzzy robustness index	38
3.	3.6	Fuzzy resiliency index	40
4	AN	ALYSIS OF THE LAKE HURON SYSTEM	43
4.1	L	HPWSS system representation and data	43
4.2	R	lesults	43
4.	2.1	Assessment of the fuzzy performance Indic ices	43
4.	2.2	Importance of different membership function shapes	48
4.	2.3	Significance of system components	51
5	AN	ALYSIS OF THE ELGIN AREA SYSTEM	54
5.1	E	APWSS system representation and data	54
5.2	R	Results	54
5.	2.1	Assessment of the fuzzy performance Indic ices	55
5.	2.2	Importance of different membership function shapes	55
5.	2.3	Significance of system components	59
6	CO	NCLUSIONS	62
7	RE	FERENCE	66

LIST OF FIGURES

Figure (1): The city of London regional water supply system	4
Figure (2) Schematic representation of the LHPWSS	6
Figure (3) Schematic representation of the EAPWSS	10
Figure (4) Water supply system layout.	14
Figure (5) System integrated layout for the reliability analysis calculation	15
Figure (6) Support and a-cut of the fuzzy membership function	18
Figure (7) Membership function development using design capacity and overload	
capacity	21
Figure (8) Supply requirement membership function	22
Figure (9) Input data for LHPWSS	28
Figure (10) Fuzzy representation of the acceptable failure region	29
Figure (11) Typical example of a multi-component system	33
Figure (12) Calculation of the system-state membership function for the multi-com	ponent
system	34
Figure (13) Overlap area between the system-state membership function and the	
acceptable level of performance	35
Figure (14) Flow chart for the fuzzy reliability-vulnerability index calculation	37
Figure (15) Flow chart for the fuzzy robustness index calculation	39
Figure (16) Flow chart for calculation of the fuzzy resiliency index	42
Figure (17) LHPWSS system integrated layout- part 1	44
Figure (17) LHPWSS system integrated layout- part 2	45
Figure (17) LHPWSS system integrated layout- part 3	46
Figure (18) Acceptable levels of performance	47
Figure (19) Resulting system-state membership functions for triangular and trapezo	oidal
input membership functions	50
Figure (20) System-failure membership functions using triangular and trapezoidal	shapes.
	50

Figure (21) System-state membership function change for different system components.
52
Figure (22) EAPWSS system integrated layout- part 156
Figure (22) EAPWSS system integrated layout- part 257
Figure (23) Resulting EAPWSS system-state membership functions for triangular and
trapezoidal input membership functions59
Figure (24) System-state membership function change with introduction of system
components60
Figure (25a) Fuzzy performance indices for the LHPWSSS and EAPWSS systems for the
triangular membership function shape63
Figure (25b) Fuzzy performance indices for the LHPWSSS and EAPWSS systems for the
trapezoidal membership function shape63

LIST OF TABELS

Table (1) MF calculations for a multi-component system.	-33
Table (2) The LHPWSS system fuzzy performance indices for different membership	
function shapes	-49
Table (3) System fuzzy performance indices change due to the improvement of PAC	
transfer pump capacity	-52
Table (4) Change in the system fuzzy performance indices due to change in the maxim	um
capacity of the pac transfer pump	-54
Table (5) The EAPWSS system fuzzy performance indices for different membership	
function shapes	-58
Table (6) System fuzzy performance indices change due to the change in the PAC	
maximum capacity	-61

1 INTRODUCTION

1.1 Objectives of the analysis

This study explores the utility of fuzzy performance indices: (i) combined reliabilityvulnerability index, (ii) robustness index, and (iii) resiliency index, for evaluating the performance of a complex water supply system. Regional water supply system for the City of London is used as the case study. The two main components being investigated in this case study are; (i) the Lake Huron Primary Water Supply System (LHPWSS), and (ii) the Elgin Area Primary Water Supply system (EAPWSS).

Computational requirements for the implementation of the fuzzy performance indices are investigated together with the sensitivity of these criteria to different shapes of fuzzy membership functions.

1.2 Report organization

Chapter 2 briefly introduces the Lake Huron Primary Water Supply System (LHPWSS), and the Elgin Area Primary Water Supply system (EAPWSS). Chapter 3 presents the methodology used for the analysis of both systems. The chapter starts by describing the procedure for system representation. The description of the method used to construct membership functions for different system components follows. The calculation process of the fuzzy performance indices is presented in details at the end.

Chapters 4 and 5 present the fuzzy performance indices for LHPWSS and EAPWSS systems, respectively. In both chapters, the sensitivity of fuzzy indices to the different shapes of fuzzy membership functions is explored first. The utility of these measures in identifying critical system components is demonstrated afterwards. Finally, the conclusions of **t**he analysis performed in the previous two chapters are presented in Chapter 6.

1.3 Summary of the results

The analysis of the results revealed that LHPWSS system is reliable and not too vulnerable to disruption in service. On the contrary, EAPWSS system is found to be highly unreliable and vulnerable to disruption in service. The results show that LHPWSS system is more robust than EAPWSS system, and therefore LHPWSS system can accommodate possible change in requirement conditions.

Combined reliability-vulnerability index and robustness index are sensitive to change in the shape of the membership function. The value of the resiliency index does not depend on the shape of membership function.

The fuzzy performance indices are capable of identifying weak system components that require attention in order to achieve future improvement in system performance.

2 SYSTEM DESCRIPTION

The City of London regional water supply system consists of two main components; (i) the Lake Huron Primary Water Supply System (LHPWSS), and (ii) the Elgin Area Primary Water Supply system (EAPWSS). The LHPWSS system obtains raw water from the Lake Huron. Water is treated and pumped from the lake to the terminal reservoir in Arva, as shown in Figure 1. Water from the Arva reservoir is pumped to the north of the City of London where it enters the municipal distribution system. The system provides water for the City of London as well as a number of smaller neighboring municipalities (through a secondary system).

The EAPWSS system treats raw water from the Lake Erie and pumps the treated water to the terminal reservoir located in St. Thomas. Water from the reservoir is pumped to the south of the City of London where it enters the municipal distribution system, as shown in Figure 1. In the case of emergency, the City of London can obtain additional water from a number of wells located inside the City and in the surrounding areas.

2.1 Lake Huron primary water supply system (LHPWSS)

The Lake Huron treatment facility has a treatment capacity of about 336 million liters per day (336,400 m³/day). The plant's individual components are designed with a 35% overload capacity resulting in the maximum capacity of 454,600 m³/day. The current daily production, based on the annual average, is 157,000 m³/day with a maximum production value of 264,000 m³/day in 2001.



Figure (1): The City of London regional water supply system.

The water treatment system employs conventional and chemically assisted flocculation and sedimentation systems, dual-media filtration, and chlorination as the primary disinfection. Both, the treatment system and the water quality are continuously monitored using computerized Supervisor Control and Data Acquisition (SCADA) system.

A brief description of the system's works, from the intake through the treatment plant to the terminal reservoir at Arva is provided in the following section. A schematic representation of the system is depicted in Figure 2.

2.1.1 Intake system

Raw water flows by gravity from Lake Huron through a reinforced concrete intake pipe to the low lift pumping station. The intake pipe discharges raw water through mechanically cleaned screens into the pump-well of the low lift pumping station. The intake crib and the intake pipe are designed for the maximum capacity of 454,600 m^3 /day. Chlorine can be injected in the intake crib through the screens or to the low lift pumping station for zebra mussel control (pre-chlorination). The low lift pumping station is located on the shore of Lake Huron at the treatment plant site. The low lift pumping station consists of six pumps with rated capacity between 115,000 and 100,000 m^3 /day.



Figure (2) Schematic representation of the LHPWSS.

2.1.2 Water treatment system

Water from the bw lift pumping station is discharged into the treatment plant where it bifurcates into two parallel streams designated as the North and the South. Two flash mix chambers, one in each stream, consist of two cells and one mixer per cell. The water flows by gravity from the flash mix chambers to the flocculation tanks.

In the first treatment step, which takes place in the flash mix chambers, Alum is added (for coagulation) together with Powdered Activated Carbon (PAC) (seasonally added for taste and odor control) and Polymer (as coagulant aid). Chlorine, which is used for disinfection, is added upstream of the flash mixers.

Mechanical flocculation process takes place in both, North and South treatment lines. Each flocculation tank is divided into two zones, primary and secondary, with the capacity ranging between $32,000 \text{ m}^3/\text{day}$ and $170,000 \text{ m}^3/\text{day}$. Water flows through the two zones where walking beams (or paddle mixers) perform the mixing, to the darifiers/settlers. Water flows into the settlers from one end, flows up through the parallel plate clarifiers and is discharged at the opposite end. A scraper, at the bottom of the tank, thickens the settled solids and moves them to the central hopper.

Waste sludge pumps transfer settled solids to the solid bowl centrifuges for dewatering. The solid wastes are stored into a container for off-site disposal while the concentrate is returned to the lake through the main plant drain. Twelve high rate gravity filters perform the removal of particulate matter from water flowing from the clarifiers. Water flows to any of the twelve filters from both treatment lines. Filtered water is then discharged into the three clear-wells where Chlorine is added for post-chlorination.

2.1.3 Conveyance and storage systems

Finished water is pumped from the clear-wells through the transmission main to the terminal reservoir at Arva by the high lift pumps. The high lift pumping station consists of five high lift pumps rated at 1,158 L/s. Water flows through the primary transmission main, a 1220 mm diameter concrete pipe, under pressure for about 47 km. A total of 21 km of the primary transmission main is twined to maintain the capacity and increase the redundancy in case of emergency. The primary transmission main is surge-protected during power failure or transit pressure conditions (due to cycling of the high lift pumps). The terminal reservoir at Arva consists of four individual cells, each of 27,000m³ storage capacity.

An intermediate reservoir and booster station are constructed in the McGillivary township. The intermediate reservoir serves the users in the McGillivary township. Water from the reservoir can be withdrawn back into the primary transmission main during the high demand periods, by four high lift pumps at the booster station.

2.2 Elgin area primary water supply system (EAPWSS)

The Elgin water treatment facility was constructed in 1969 to supply water from the Lake Erie to the City of London, St. Thomas and a number of smaller municipalities. In 1994, the facility has been expanded to double its throughput to its current $91,000m^3/day$ capacity. A series of upgrades took place from 1994 to 2003 to add surge protection and introduce fluoridation treatment. The design capacity of the treatment facility is $91,000m^3/day$ m³/day, with an average daily flow of $52,350 m^3/day$, which serves about 94,400 persons.

The water treatment in EAPWSS employs almost the same conventional treatment methods used in LHPWSS. The only exception is that the facility uses the fluoridation treatment system to provide dental cavity control to the users. As in LHPWSS, the treatment system and water quality are continuously monitored using computerized Supervisor Control and Data Acquisition (SCADA) system. The finished treated water is pumped to the terminal reservoir located in St. Thomas. A short description of the EAPWSS is given in the following section. A schematic of the system is shown in Figure 3.



Figure (3) Schematic representation of the EAPWSS.

2.2.1 Intake system

Raw water, drawn from the Lake Erie, is pumped through a 1500 mm diameter intake conduit to the low lift pumping station at the shore of the lake. The ultimate capacity of the intake conduit is $182,000 \text{ m}^3/\text{day}$; in case of an emergency the plant drain serves as an alternative intake, with almost the same maximum capacity. The low lift pumping station houses two clear-wells. Each well has two independent vertical turbine pumps that discharge into a 750 mm transmission main to the water treatment plant.

2.2.2 Water treatment system

The raw water discharged from the low lift pumping station is metered and split evenly into two parallel streams, as in the LHPWSS. The split continues from the head-works to the filtration process. The first treatment process is the flash mixing where Alum is added as a coagulation agent together with PAC. There is one flash mixing chamber with two cells and one mixer per cell in each treatment line. Water flows by gravity from the flash mix chamber to the flocculation tanks.

The flocculation system consists of two banks, North and South, of flocculation tanks, each with a capacity of 91,000 m^3/day . Each bank has two tanks that make a total of eight flocculation tanks. Polymer can be added at any point in the series of flocculation tanks. Water flows directly from the flocculation tanks into the sedimentation system. There is one gravity sedimentation tank in each process stream. Pre-chlorination takes place after the sedimentation process and before the filtration.

Finally, the particulate matter is removed using four gravity filters during the filtration process. The treatment is no longer split into two parallel streams as the water can be directed to any of the four filters. The filtered water is collected in the filtered water conduit underlying the filters and flows into a clear well and the on-site reservoir. Post-chlorination takes place in the conduit leading from the on-site reservoir to the high lift pumping station.

2.2.3 Conveyance and storage systems

The high lift pumping station delivers finished water through the transmission main to the terminal reservoir in St. Thomas. It also delivers water to the secondary distribution system. The high lift pumping station houses four high lift pumps, each with a rated capacity of 52,000 m^3 /day. The treated water is discharged through the primary transmission main (14 km long 750 mm diameter concrete pressure pipe).

The surge facility was constructed in 1994 to protect the transmission main from damage due to the system transit pressure conditions during cycling of the high lift pumps. Through the valve chamber, upstream of the terminal reservoir, water from the transmission main is directed to one, or both, reservoirs at the Elgin-Middlesex facility. Both reservoirs have equal capacity of 27,300 m³ and store water supply for Aylmer, St. Thomas and the Elgin-Middlesex (serving London) pumping system. Water can by-pass the reservoirs and flow directly to each of the secondary pumping stations.

3 METHODOLOGY FOR SYSTEM RELAIBILITY ANALYSIS

3.1 Multi-component system representation

Water supply system is a typical example of a multi-component system that includes a collection of conveyance, treatment, and storage components. These components are at risk of failure due to a wide range of causes. In the same time, these elements are connected in complicated networks that affect the overall performance of the water supply system.

The key step in the evaluation of system performance is the appropriate representation of different relationships between system components. This representation should reflect the effect of the performance of each component on the overall system performance. For example, the chemical treatment of raw water in a water supply system depends on adding different chemicals at certain locations in the treatment process. This process requires the availability of chemicals in the storage facility and the ability to transfer them to the required location on time. Storage and conveyance facilities, responsible for delivering these chemicals to the mixing chambers, are not part of the raw water path. The failure of the set facilities directly affects the water treatment process and might cause a total failure of the water treatment system. As a result, it is important to consider these facilities when performing a system reliability analysis.

Figure 4 shows the layout of one part of the water treatment plant, where the stored chemicals are conveyed to the mixing location via the feed pump. It is evident that taking

these components into consideration in the system reliability analysis is difficult becuase of the need to identify the functional relationships between them and the other system components. Similar relationships are required for all non-carrying water components. If these components are not taken into consideration the chance of improper estimation of system reliability may increase.



Figure (4) Water supply system layout.

Representing a multi-component system as a system of components having different failure relationships can be used as an effective mean to integrate water-carrying and non-water carrying components into one system. For example, any two components are considered serially connected if the failure of one component leads to the failure of the other. Two components are considered to have a parallel connection if the failure of one component does not lead to the failure of the other. A clear identification of the failure relationship between different components facilitates the calculation of the performance indices. Figure

5 shows the integrated layout for the previous example. In this figure, the system representation integrates components carrying chemicals into the path of raw water.

Calculation of the system's performance indices based on the integrated layout will be fairly difficult as there is no clear link between the failure of the components carrying chemicals and the components carrying raw water. Note that querational components having redundancy are treated as components with parallel connection. This reflects the fact that redundant elements reduce the possibility of system failure.



Figure (5) System integrated layout for the reliability analysis calculation.

3.2 Capacity and requirement of system components

System reliability analysis uses load and resistance as the fundamental concepts to define the risk of system failure, (Simonovic, 1997). These two concepts are used in structural engineering to reflect the characteristic behavior of the system under external loading conditions. In water supply systems, load and resistance are replaced by requirement and capacity, respectively, to reflect the specific domain variables of the water supply system. Hence, system requirement is defined as the variable that reflects different water demand requirements that may be imposed over the useful life of the system (Ang and Tang, 1984). System capacity, on the other hand, is defined as the system characteristic variable which describes the capacity of the system to satisfy demand requirements.

The fuzzy reliability analysis uses membership functions (MFs) to express uncertainty in both capacity and requirement of each system component. The general representation of membership function is:

$$\tilde{X} = \{(x, \mu_{\tilde{X}}(x)) : x \in \mathbb{R}; \mu_{\tilde{X}}(x) \in [0,1]\}$$
(1)

where:

- \tilde{X} is the fuzzy membership function;
- $\mu_{\tilde{x}}(x)$ is the membership value of element *x* to \tilde{X} ; and
- R is the set of real numbers.

Membership functions are usually defined by their α -cuts. The α -cut is the ordinary set of all the elements belonging to the fuzzy set whose value of membership is a or higher, that is:

$$X(a) = \{ x : \mu_{\tilde{X}}(x) \ge a ; x \in R; a \in [0,1] \}$$
(2)

where

X(a) is the ordinary set at the a-cut; and

a is the membership value.

Another characteristic property of the fuzzy membership function is its support. The support of the fuzzy membership function can be defined as the ordinary set that is:

$$S(\tilde{X}) = \tilde{X}(0) = \{x : \mu_{\tilde{x}}(x) > 0\}$$
(3)

where

 $S(\tilde{X})$ is the ordinary set at the a-cut=0.

The fuzzy membership function support is the 0-cut set and includes all the elements with the membership value higher than 0, as shown in Figure 6. Construction of membership function is based on the system design data and choice of the suitable shape. There are many shapes of membership functions. However, the application context dictates the choice of the suitable shape. For the problem domain addressed in this study, system components have maximum and minimum capacity that cannot be exceeded. Therefore, any candidate membership function shape should have two extreme bounds with zero membership values. Triangular and trapezoidal shapes are the simplest MF shapes that meet this requirement.



Figure (6) Support and a-cut of the fuzzy membership function (after Ganoulis, 1994).

In the presented case study, the following reports are used as the source of data for determining capacity and requirement for each component:

- Earth Tech Canada Inc.,2000;
- Earth Tech Canada Inc.,2001;
- o American Water Services Canada-AWSC, 2003a;
- o American Water Services Canada-AWSC, 2003b; and
- DeSousa and Simonovic, 2003.

Some problems are experienced with the available data. First, many components have single design capacity that creates a problem in the development of a membership function. The second problem is the use of different units for capacity of different components. For instance, capacity of storage facilities is expressed in volumetric units, cubic meters (m^3) . Capacity of pumps is measured using flow units, cubic meter per day (m^3/day) . Thus, their

direct comparison may not be possible. The third problem is the identification of the requirement for each system component. Most of the available information corresponds to the system requirement (i.e., the requirement of the chlorination system not the capacity of individual chlorinator).

3.2.1 System component capacity membership function

A triangular membership function, representing the capacity of a system component, is constructed using three design values (i.e., the minimum, modal, and the maximum value). In many cases only one value is available. For example in the case of reservoirs, only the maximum capacity is available. If there is no other source of information, the minimum capacity is set to zero. The modal value can be subjectively selected within the range from minimum to maximum capacity. In case of trapezoidal membership function, two modal points are subjectively selected.

In cases when the components are designed with an overload capacity (i.e. maximum design capacity higher than the rated capacity) this value is used to build the membership function. Figure 7 depicts a component with a maximum capacity of a units with c (%) overload capacity. In case (I), a triangular membership function is defined as follows:

$$\mu_{\bar{A}}(x) = \begin{cases} 0, & \text{if } x \le (1-2c)a \\ \frac{x - (1-2c)a}{(1-c)a - (1-2c)a}, & \text{if } x \in [(1-2c)a, (1-c)a] \\ \frac{a - x}{a - (1-c)a}, & \text{if } x \in [(1-c)a, a] \\ 0, & \text{if } x \ge a \end{cases}$$
(4)

where

(1-c)a is the modal value; and

(1-2c)a and a are the lower and upper bounds of the membership function.

In case (II), a trapezoidal membership function is defined as follows:

$$\mu_{\tilde{A}}(x) = \begin{cases} 0, & \text{if } x \le (1-2c)a \\ \frac{x - (1-2c)a}{(1-1.5c)a - (1-2c)a}, & \text{if } x \in [(1-2c)a, (1-1.5c)a] \\ 1, & \text{if } x \in [(1-2c)a, (1-1.5c)a] \\ \frac{1}{a - x}, & \text{if } x \in [(1-1.5c)a, (1-0.5c)a] \\ \frac{1}{a - (1-0.5c)a}, & \text{if } x \in [(1-0.5c)a, a] \\ 0, & \text{if } x \ge a \end{cases}$$

where

(1-1.5c)a and (1-0.5c)a are the modal values; and

(1-2c)a and a are the lower and upper bounds of the membership function.

The modal values in case (II) (i.e. trapezoidal membership function) equally divide the distance from the modal value (in the triangular membership function) to the lower and upper bounds, respectively. In both cases the maximum value corresponds to the design capacity.



Figure (7) Membership function development using design capacity and overload capacity.

3.2.2 System component requirement membership function

The requirement membership function of a group of components performing the same function is based on the assumption of equal role for every unit. For example, if a collection of four chlorinators supply Y [kg/day] of Chlorine, the maximum supply requirement of each chlorinator is (Y/4) [kg/day]. The yearly average and minimum

requirements are used to develop the requirement membership function of each component, as shown in Figure 8.



Figure (8) Supply requirement membership function.

The two modal values of the trapezoidal membership function in Figure 8 are the middle points between the maximum, or minimum, supply and the average requirement value. In case when yearly average data is not available, the modal value is considered to be the average value of the maximum and minimum supply.

Proxy conversions are used to overcome the problem of using different units for expressing capacity and requirement. For example, the supply requirement of certain chemical is usually expressed in kilograms (kg) while the storage facility capacity is expressed in cubic meters (m³). In this case, the corresponding chemical bulk density is used to convert the supply requirement using volumetric units.

3.2.3 Standardization of membership functions

In the process of calculating system fuzzy reliability indices, membership functions of system components are aggregated using fuzzy operators. Therefore, all membership functions must be expressed in the same units. This can be achieved only through standardization of the membership functions (i.e., division by the unit maximum capacity value).

The membership function of each system component will have a maximum value of one. For example, a triangular membership function representing a reservoir capacity (m³) is defined as follows:

where

m is the modal value; and

a and b are the lower and upper bounds of the non-zero values of the membership.

This membership function is standardized to the following (dimensionless) membership function:

$$\mu_{\bar{A}}(x) = \begin{cases} 0, & \text{if } x \le (a/b) \\ \frac{x - (a/b)}{(m/b) - (a/b)}, & \text{if } x \in [(a/b), (m/b)] \\ \frac{1 - x}{1 - (m/b)}, & \text{if } x \in [(m/b), 1] \\ 0, & \text{if } x \ge 1 \end{cases}$$
.....(7)

where

(m/b) is the modal value; and

(a/b) and 1 are the lower and upper bounds of the non-zero

values of the membership.

The capacity and requirement membership functions are processed together as one membership function representing the component-state membership function. The same standardization method is applied to the requirement membership functions. The membership function values are divided by the maximum capacity of a system component.

3.3 Calculation of fuzzy performance indices

The membership functions representing system-state and acceptable levels of performance are used in the calculation of the fuzzy reliability-vulnerability and robustness indices.

3.3.1 System-state membership function

Multi-component systems have several component-state membership functions describing each component of the system. Aggregation of these membership functions results in the system-state membership function for the whole-system (El-Baroudy and Simonovic, 2003 and 2004).

First, all parallel and redundant components are aggregated into a number of serially connected components. For a group of M parallel (or redundant) components, the m th component has a component-state membership function $\tilde{S}_m(u)$ defined on the universe of discourse U. All the components states contribute to the whole group system-state membership function. Failure of the group occurs if all components fail. Hence, the system-state is calculated as follows:

where:

 \tilde{S}_m (u) is the m-th component-state membership function; and

M is the total number of parallel (or redundant) components.

For the system of N serially connected groups, where the *n*-th group has a state membership function $\tilde{S}_n(u)$, the weakest component controls the whole system-state or causes the failure of the whole system. Therefore, the system-state is calculated as follows:

where:

 $\Tilde{S}(u)$ is the system-state membership function; and

 $\left(\tilde{S}_{l},\tilde{S}_{2},...,\tilde{S}_{N}\right)$ component-state membership functions.

In the present case study, all component-state membership functions are formulated in terms of fuzzy margin of safety using the fuzzy subtraction operator (El-Baroudy and Simonovic, 2003 and 2004).

where:

 \tilde{M}_i is the fuzzy margin of safety of the i-th component; \tilde{X}_i is the fuzzy capacity of the i-th component; \tilde{Y}_i is the fuzzy requirement of the i-th component; and

n is the number of system components.

Capacity and requirement membership functions are stored in the spreadsheet, where all the necessary calculations are performed to obtain the final component-state and component-failure membership functions. Figure 9 shows a part of the spreadsheet for LHPWSS, while Appendix (I) contains the full-length spreadsheet files for both systems under

investigation (LHWPSS and EAPWSS). The fuzzy performance indices are then calculated using the calculation script that is developed to perform different calculation steps. Appendix (II) includes the source code of the script files for both LHPWSS and EAPWSS.

3.3.2 Acceptable level of performance membership function

The acceptable level of performance is a fuzzy membership function that is used to reflect the decision-makers ambiguous and imprecise perception of risk, (El-Baroudy and Simonovic, 2003 and 2004). The reliability reflected by the acceptable level of performance is quantified by

where:

LR is the reliability measure of the acceptable level of performance; and

 x_1 and x_2 are the bounds of the acceptable failure region, as shown in Figure 10.

The calculation of the fuzzy reliability-vulnerability and fuzzy robustness indices depends on the calculation of the overlap area between the membership functions of both the system-state and the acceptable level of performance.

2	Microsoft Excel - Case Study-	Calculation(LH).xls						_ 8 ×		
] <u>F</u> ile <u>E</u> dit ⊻iew <u>I</u> nsert F <u>o</u> rr	nat <u>I</u> ools <u>D</u> at	<u>I</u> ools <u>D</u> ata <u>W</u> indow <u>H</u> elp					Type a question for help 🛛 👻 🗕 🗗 🗲			
	Arial		16 - B	1 日 華		\$ %,	÷.00 .00 €	E (E	3 - A -		
E) 🗢 🖬 a 🖬 a 🛯 🗢 🕻	1 × Pa ma -	0 m -	E.Σ	- 21 21	80%	• 2 -				
	A4 - 5										
1	A	Ð	Н	1	J	K	L	M	N T		
4		System	Units		Capacity		Requirment				
5				Average Daily flow	Design Capacity	Maximum Overload	Yearly Min	Yearly Avg	Yearly Max		
6 7 8	Intake Crib Chlorinator I RC Intake Pipe	Intak e Syste m	MLD Kg/d MLD	157.3 360.0 157.3	340.0 630.0 340.0	454.6 900.0 454.6	53.0 24.5 53.0	157.3 72.0 157.3	255.7 130.0 255.7		
9 10 11 12 13 14 15 16 17	Traveling Screens Pumping Wells Chlorinator II Single Speed Pump 1 Variable Speed Pump 2 Single Speed Pump 1 (Back-up Variable Speed Pump 2 (Back-up Single Speed Pump 2 (Back-up	Low Lifting System	MLD MLD Kg/d MLD MLD MLD MLD MLD MLD	157.3 157.3 360.0 49.9 57.4 49.9 49.9 57.4 49.9	340.0 340.0 630.0 75.0 86.2 75.0 75.0 86.2 75.0 75.0	454.6 454.6 900.0 100.0 115.0 100.0 100.0 115.0 100.0	53.0 53.0 24.5 16.8 19.3 16.8 19.3 16.8 19.3 16.8	157.3 157.3 72.0 49.0 49.0 49.0 49.0 49.0 49.0 49.0	255.7 255.7 130.0 81.2 93.4 81.2 81.2 93.4 81.2 93.4 81.2		

Elemen	+ Canto (7	And a second sec								i inter	Sector Sector		OF:
	t-state (1	ri-MoS)	Elen	Element-State (Trap-MoS)			Element-Failure (Tri)			Element-Failure (Trap)			
а	b	o	ð	4	¢	d	а	ħ	¢	8	ð	c	đ
-0.22	0.40	0.88	-0.22	0.09	0.41	0.88	0.08	0.13	0.17	0.08	0.11	0.14	0.17
0.26	0.62	0.97	0.26	0.44	0.74	0.97	0.01	0.01	0.01	0.01	0.01	0.01	0.01
-0.22	0.40	0.88	-0.22	0.09	0.41	0.88	0.08	0.13	0.17	0.08	0.11	0.14	0.17
-0.22	0.40	0.88	-0.22	0.09	0.41	0.88	0.04	0.08	0.50	0.04	0.19	0.35	0.50
-0.22	0,40	0.88	-0.22	0.03	0.41	0.88	0.00	0.50	1.00	0.00	0.33	0,67	1.00
0,26	0.62	0.97	0.26	0.44	0.74	0.97	0.01	0.01	0.01	0.01	0.01	0.01	0.01
-0.31	0.26	0.83	-0.31	-0.03	0.22	0.83	0.02	2.51	5.00	0.02	1.68	3.34	5.00
-0.31	0.32	0.83	-0.31	0.01	0.32	0.83	0.02	2.51	5.00	0.02	1.68	3.34	5.00
-0.31	0.26	0.83	-0.31	-0.03	0.22	0.83	0.02	2.51	5.00	0.02	1.68	3,34	5.00
-0.31	0.26	0.83	-0.31	-0.03	0.22	0.83	0.02	2.51	5.00	0.02	1.68	3,34	5.00
-0.31	0.32	0.83	-0.31	0.01	0.32	0.83	0.02	2.51	5.00	0.02	1.68	3.34	5.00
-0.31	0.26	0.83	-0.31	-0.03	0.22	0.83	0.02	2.51	5.00	0.02	1.68	3.34	5.00
	9 -0.22 -0.22 -0.22 -0.22 -0.22 -0.26 -0.31 -0.31 -0.31 -0.31 -0.31 -0.31	a b -0.22 0.40 0.26 0.62 0.22 0.40 -0.22 0.40 -0.22 0.40 -0.22 0.40 -0.26 0.62 -0.31 0.26 -0.31 0.26 -0.31 0.26 -0.31 0.26 -0.31 0.26 -0.31 0.26 -0.31 0.26	s h c -0.22 0.40 0.88 0.26 0.62 0.81 -0.22 0.40 0.88 -0.22 0.40 0.88 -0.22 0.40 0.88 -0.22 0.40 0.88 -0.26 0.62 0.97 -0.31 0.26 0.83 -0.31 0.32 0.83 -0.31 0.26 0.83 -0.31 0.26 0.83 -0.31 0.26 0.83 -0.31 0.26 0.83 -0.31 0.26 0.83 -0.31 0.26 0.83 -0.31 0.26 0.83	a b c s 0.22 0.40 0.85 0.22 0.26 0.62 0.97 0.26 0.22 0.40 0.85 -0.22 0.22 0.40 0.85 -0.22 -0.22 0.40 0.86 -0.22 -0.22 0.40 0.86 -0.22 -0.22 0.40 0.86 -0.22 -0.26 0.62 0.31 0.26 -0.26 0.62 0.31 0.26 -0.31 0.26 0.83 -0.31 -0.31 0.26 0.83 -0.31 -0.31 0.26 0.83 -0.31 -0.31 0.26 0.83 -0.31 -0.31 0.26 0.83 -0.31 -0.31 0.26 0.83 -0.31 -0.31 0.26 0.83 -0.31 -0.31 0.26 0.83 -0.31	s h c s h 0.22 0.40 0.88 0.22 0.09 0.26 0.62 0.81 0.26 0.44 0.22 0.40 0.88 0.22 0.09 0.22 0.40 0.88 0.22 0.09 0.22 0.40 0.88 0.22 0.09 0.22 0.40 0.88 0.22 0.09 0.26 0.62 0.31 0.26 0.44 0.31 0.26 0.83 -0.31 -0.03 -0.31 0.26 0.83 -0.31 0.01 -0.31 0.26 0.83 -0.31 0.03 -0.31 0.26 0.83 -0.31 0.03 -0.31 0.26 0.83 -0.31 -0.03 -0.31 0.26 0.83 -0.31 -0.03 -0.31 0.26 0.83 -0.31 0.01	a h c s h c 0.22 0.40 0.88 0.22 0.09 0.41 0.26 0.62 0.37 0.26 0.44 0.74 0.22 0.40 0.68 -0.22 0.09 0.41 0.22 0.40 0.68 -0.22 0.03 0.41 -0.22 0.40 0.68 -0.22 0.03 0.41 -0.22 0.40 0.68 -0.22 0.09 0.41 -0.22 0.40 0.88 -0.22 0.09 0.41 -0.22 0.40 0.88 -0.22 0.09 0.41 -0.24 0.40 0.88 -0.22 0.09 0.41 0.26 0.62 0.37 0.26 0.44 0.74 -0.31 0.26 0.83 -0.31 -0.03 0.22 -0.31 0.26 0.83 -0.31 -0.03 0.22 -0.31 0.26 0.83	s h c s h c d 0.22 0.40 0.88 0.22 0.09 0.41 0.86 0.22 0.40 0.88 0.22 0.09 0.41 0.86 0.22 0.40 0.88 -0.22 0.09 0.41 0.86 0.22 0.40 0.88 -0.22 0.09 0.41 0.86 -0.22 0.40 0.88 -0.22 0.09 0.41 0.86 -0.22 0.40 0.88 -0.22 0.09 0.41 0.86 -0.22 0.40 0.88 -0.22 0.09 0.41 0.86 -0.24 0.40 0.88 -0.22 0.09 0.41 0.86 -0.25 0.40 0.88 -0.22 0.09 0.41 0.86 -0.25 0.40 0.83 -0.31 0.32 0.83 -0.31 0.26 0.83 -0.31 0.01 0.32 0.83 <	$\begin{array}{c c c c c c c c c c c c c c c c c c c $	$\begin{array}{c c c c c c c c c c c c c c c c c c c $	$\begin{array}{c c c c c c c c c c c c c c c c c c c $	$\begin{array}{c c c c c c c c c c c c c c c c c c c $	s h c s s h c s	s h c s s

Figure (9) Input data for LHPWSS



Figure (10) Fuzzy representation of the acceptable failure region.

3.3.3 System-failure membership function

The system-failure membership function is used in the calculation of the fuzzy resiliency index. This membership function represents the system's time of recovery from the failure state. For each type of failure the system might have a different recovery time. Therefore, a series of fuzzy sets, each for different type of failure, are developed for the system under consideration (El-Baroudy and Simonovic, 2003 and 2004). Then the maximum recovery time is used to represent the system-characteristic recovery time as follows, (Kaufmann and Gupta, 1985)
where:

 $\tilde{T}(a)$ is the system fuzzy maximum recovery time at α -cut (as defined by Equation 2); $t_{1_j}(a)$ is the lower bound of the j-th recovery time at α -cut (as defined by Equation 2); $t_{2_j}(a)$ is the upper bound of the j-th recovery time at α -cut

(as defined by Equation 2); and

J is total number of fuzzy recovery times.

Multi-component systems have several system-failure membership functions representing the system-failure for each component. Aggregation of these membership functions results in a system-failure membership function for the whole-system.

Parallel and redundant components are aggregated into serial groups using the fuzzy maximum operator. For parallel system configuration composed of M components, the m-th component has a maximum recovery time membership function $\tilde{T}_m(t)$, defined on the universe of discourse T. Therefore, the system-failure membership function (i.e. the membership function that represents the system recovery time) can be calculated as follows, (El-Baroudy and Simonovic, 2003 and 2004)

 where:

 $\tilde{T}(t)$ is the whole system-failure membership function; and $(\tilde{T}_{p}, \tilde{T}_{2}, \dots, \tilde{T}_{M})$ are recovery time membership functions for different components.

The system-failure membership function is then calculated for the N serially connected components using, (El-Baroudy and Simonovic, 2003 and 2004)

given

$$\mathbf{S}(\tilde{\mathbf{T}}_{c}) = \max_{N} \left(\mathbf{S}(\tilde{\mathbf{T}}_{1}), \mathbf{S}(\tilde{\mathbf{T}}_{2}), \dots, \mathbf{S}(\tilde{\mathbf{T}}_{N}) \right)$$

and

.....(15)

$$\tilde{T}_{c}(1) = \max_{N} \left(\tilde{T}_{1}(1), \tilde{T}_{2}(1), \dots, \tilde{T}_{N}(1) \right)$$

where:

 $\tilde{T}(t)$ is the whole system recovery time membership function; $\tilde{T}_c(t)$ is the controlling recovery time membership function; $S(\tilde{T}_c)$ is the support of the controlling recovery time membership function (as defined by Equation 3); $(S(\tilde{T}_1), S(\tilde{T}_2), \dots, S(\tilde{T}_N))$ are the support of the N components recovery time membership functions (as defined by Equation 3); $\tilde{T}_c(1)$ is the controlling recovery time membership function at the α -cut =1 (as defined by Equation 2); and $(\tilde{T}_1(1), \tilde{T}_2(1), \dots, \tilde{T}_N(1))$ are the recovery time membership functions

at α -cut =1 of the N components (as defined by Equation 2).

3.3.4 Fuzzy reliability-vulnerability index

Figure 11 shows an example of a multi-component system. The system has two parallel components connected serially to a third component that has a redundant component. The component-state membership functions for all five components are listed in Table 1, together with the system-state membership functions for the parallel and redundant components.

Figure 12 illustrates the process of calculating the system-state membership function for the given example. The membership functions of parallel and redundant components are summed to obtain three system-state membership functions for the serial components. The resulting membership function is then calculated using the fuzzy minimum operator, represented by the shaded area in Figure 12.



Figure (11) Typical example of a multi-component system.

Component	MF	Parallel Summation	Redundant Summation	System-State MF
Component 1	(1,2,3)	(2,4,6)	NA	
Component 2	(1,2,3)		NA	Min [(2,4,6), (1,3,5),
Component 3	(1,3,5)	NA	NA	(1,2,3)]
Component 4	(0.5,1,1.5)	NA	(1,2,3)	
Component 5	(0.5,1,1.5)	NA		

Table (1) MF calculations for a multi-component system.

The compatibility between the system-state and the acceptable level of performance is the basis for the calculation of the fuzzy combined reliability-vulnerability performance index, as shown in Figure 13.



Figure (12) Calculation of the system-state membership function for the multi-component system.

The compatibility measure (CM) is calculated as:

$$CompatibilityMeasure (CM) = \frac{Weightedoverlaparea}{Weightedareaofsystem-statefunction}$$
.....(16)

and then used to calculate the combined fuzzy reliability-vulnerability performance index

where:

 LR_{max} is the reliability measure of acceptable level of performance for which the system-state has the maximum compatibility value(CM);

LR_i is the reliability measure of the i-th acceptable level of performance

(as defined by Equation 11);

 CM_i is the compatibility measure for system-state with the i-th acceptable level of performance; and

K is the total number of defined acceptable levels of performance.



Figure (13) Overlap area between the system-state membership function and the acceptable level of performance.

Figure 14 shows the flow chart for the calculation of the fuzzy combined reliabilityvulnerability index;

- Step (1); reading input data from the spreadsheet file containing the componentstate membership functions. Both types of membership functions, triangular and trapezoidal, are constructed;
- Step (2); storing the input data in an appropriate data format (i.e., structure array).
- Step (3); transforming input data into both, triangular and trapezoidal membership function shapes. Appendix (II) contains source code for transformation into triangular and trapezoidal shapes;
- Step (4); all parallel and redundant components are augmented using the fuzzy summation operator to calculate the membership functions representing the parallel and the redundant groups, respectively. The system is turned into a group of serially connected components, and then the maximum operator is used to calculate the system-state membership function. Appendix (II) contains the source code for the fuzzy operator, specially designed for this case study; and
- Step (5); calculating the fuzzy combined reliability-vulnerability index based on the overlap area between the system-state and the acceptable level of performance.



Figure (14) Flow chart for the fuzzy reliability-vulnerability index calculation.

3.3.5 Fuzzy robustness index

Robustness is a measure of system performance that is concerned with the ability of the system to adapt to a wide range of possible demand conditions, in the future, at little additional cost (Hashimoto et al, 1982b). The fuzzy form of change in future conditions can be reflected through the change in the acceptable level of performance and, also, in the change of the system-state membership function (El-Baroudy and Simonovic, 2003 and 2004). The change in overlap area is used to calculate system fuzzy robustness index as follows:

FuzzyRobustnessIndex=
$$\frac{1}{CM_1 - CM_2}$$
(18)

where:

CM₁ is the compatibility measure before the change in conditions; and

 CM_2 is the reliability after the change in conditions.

Figure 15 shows the flow chart for the calculation of the fuzzy robustness index;

• Step (1); reading input data from the spreadsheet file containing the componentstate membership functions. Both types of membership functions, triangular and trapezoidal, are constructed;



Figure (15) Flow chart for the fuzzy robustness index calculation.

- Step (2); storing the input data in an appropriate data format (i.e., structure array).
- Step (3); transforming input data into both, triangular and trapezoidal shapes.
 Appendix (II) contains source code for transformation into triangular and trapezoidal shapes;
- Step (4); all parallel and redundant components are augmented using the fuzzy summation operator to calculate the membership functions representing the parallel and the redundant groups, respectively. The system is transformed into a group of serially connected components, and then the maximum operator is used to calculate the system-state membership function. Appendix (II) contains the source code for the fuzzy operator, specially designed for this case study; and
- Step (5); calculating the fuzzy robustness index based on the overlap area between the system-state and predefined acceptable levels of performance.

3.3.6 Fuzzy resiliency index

Resiliency is a measure of system's time for recovery from the failure state (Hashimoto et al, 1982a). The fuzzy resiliency index is calculated using the value of the center of gravity of the system-failure membership function (El-Baroudy and Simonovic, 2003 and 2004):

FuzzyResilienceIndex=
$$\left[\frac{\int_{t_1}^{t_2} t \tilde{T}(t)dt}{\int_{t_1}^{t_2} \tilde{T}(t)dt}\right]^{-1}$$
(19)

where;

 $\tilde{T}(t)$ is the system fuzzy maximum recovery time membership function; t₁ is the lower bound of the support of the system recovery time membership function (as defined by Equation 3); and t₂ is the upper bound of the support of the system recovery time membership function (as defined by Equation 3).

The calculation script allows the use of both triangular and trapezoidal shapes, as shown in Figure 16.



Figure (16) Flow chart for calculation of the fuzzy resiliency index.

4 ANALYSIS OF THE LAKE HURON SYSTEM

4.1 LHPWSS system representation and data

The system representation provides the integrated layout that reflects the failure-driven relationships among different components. Figure 17 shows LHPWSS with all major components combined in an integrated layout. Component-state and component-failure membership functions are constructed based on the data from (Earth Tech Canada Inc.,2000), (Earth Tech Canada Inc.,2001), (American Water Services Canada-AWSC, 2003a), (American Water Services Canada-AWSC, 2003a), (American Water Services Canada-AWSC, 2003b), and (DeSousa and Simonovic, 2003) for the LHPWSS. Appendix (I) includes all the input data used in the calculation of the triangular and trapezoidal membership functions.

4.2 Results

4.2.1 Assessment of the fuzzy performance Indic ices

This section presents an assessment of the three fuzzy performance indices for the LHPWSS. Three acceptable levels of performance are arbitrary defined on the universe of the margin of safety; as (0.6,0.7,5.0,5.0), (0.6,1.2,5.0,5.0), and (0.6,5.0,5.0,5.0). They are selected to reflect three different views of decision-makers as shown by the reliability measure in Equation 11. Their reliability measures are 4.20, 1.20 and 0.68, respectively. Further, they are referred to as reliable level (level 1), neutral level (level 2), and unreliable level (level 3), as shown in Figure 18.



Figure (17) LHPWSS system integrated layout- Part 1



Figure (17) LHPWSS system integrated layout- Part 2



Figure (17) LHPWSS system integrated layout- Part 3

The results show that the combined reliability-vulnerability index for LHPWSS is 0.699. This value reflects the compatibility of the system with one of the three predefined levels of performance, as defined in Equation 17; in this case it is the reliable level (level 1). Therefore, the reliability of the system is relatively high, taking into account that the system is almost 70% compatible with the highest level of performance. The fuzzy robustness index for the LHPWSS is -2.12. Taking into consideration, that this value is the inverse of change in the overlap area, as defined in Equation 18, LHPWSS is considered to be highly robust as the overlap area increase by more than 47%. The fuzzy resiliency index value for the LHPWSS is 0.017, which means that it takes the system more than 58 days to return to the full operation mode, as defined by Equation 19. This value is relatively high as it means the system service is disrupted for about 2 months and large portion of the population served by this system (estimated to be about 325 000 person) will be affected by this disruption.



Figure (18) Acceptable levels of performance.

4.2.2 Importance of different membership function shapes

The effect of the membership function shape is investigated by calculating the three fuzzy performance indices using triangular and trapezoidal shapes. Table 2 shows the calculated fuzzy performance indices for triangular and trapezoidal membership function shapes, respectively. For the two shapes, the values of the reliability-vulnerability index are relatively high (i.e. over 0.60), taking into consideration that the maximum value of the index is 1. As shown in Figure 19, most of the system-state membership function overlaps with the reliable level of performance (level 1). This indicates that LHWPSS is highly reliable and less vulnerable to disruption in service. This is expected because; (i) the LHPWSS system has over 20 parallel groups of components and 6 redundant groups as shown in Figure 18, and (ii) many individual components are designed with a 35 % overload capacity (Earth Tech Canada Inc.,2001). This positively increases the capacity and consequently the reliability of the whole system.

There is no significant difference in the fuzzy reliability-vulnerability index values for the triangular and trapezoidal membership function shapes (i.e. the index value for the trapezoidal shape is less than 9% of the index value for the triangular shape), as shown in Table 2. This is because the change in the area of the system-state membership function is not significant and consequently the overlap area, as shown in Figure 19. Generally, it can be concluded that use of the trapezoidal shape leads to relatively lower reliability-vulnerability index than the triangular shape.

The robustness index value for LHPWSS system decreases from -2.120 (triangular shape) to -2.473 (trapezoidal shape). This corresponds to the deterioration in the system reliability-vulnerability index vale from 0.699 to 0.642. This is clear for the first case where the LHPWSS system is required to satisfy higher reliability conditions (represented by the transition from the neutral level to the reliable level). The NA values in table 2 indicate that there is no change in the overlap area and consequently the value of the robustness index will approach infinity.

The resiliency index is not affected by the shape of the membership function, since the center of gravity for both system-failure membership functions coincide, as shown in Figure 20.

Table (2) The LHPWSS system fuzzy performance indices for different membership function shapes.

Fuzzy Performance Index	Triangular MF	Trapezoidal MF
Combined Reliability-Vulnerability	0.699	0.642
Robustness (level 2 – level 1)	NA [*]	NA [*]
Robustness (level 2 – level 1)	-2.120	-2.473
Robustness (level 3 – level 2)	-2.120	-2.473
Resiliency	0.017	0.017

NA^{*} Not-available value as there is no change in overlap area.



Figure (19) Resulting system-state membership functions for triangular and trapezoidal input membership functions.



Figure (20) System-failure membership functions using triangular and trapezoidal shapes.

4.2.3 Significance of system components

System reliability depends on the reliability of its components. However, not all components are of equal importance (different location; different rate;...etc). For example, serial components have a more significant effect on the overall system reliability than parallel components, because the failure of any serial component leads to the failure of the whole system. Therefore, system's performance can only be enhanced by improving the performance of critical components. Critical component is the component that significantly reduces the area of the system-state membership function and accordingly the fuzzy performance indices of the system.

The developed computational procedure can be used to identify the critical components of the system. As mentioned in Chapter 3, the calculation transforms the multi-component system into a system of serially connected components. The fuzzy summation operator is used to turn parallel and redundant components into single entities with equivalent component-state membership functions. Then, the fuzzy minimum operator is used to sum up all serial components and entities into the system-state membership function. Observing the change in the system-state membership function can be used to identify critical system.

For the triangular membership function shape, the change resulting in the system-state membership function is shown in Figure 21. The system-state membership function changes significantly with the addition of the PAC transfer pump. This is the point where the flash mix is introduced into the system. The enhancement of flash mix system components will lead to the enhancement of the overall system performance. Looking into the components of the flash mix system, it is found that the PAC transfer pump has the smallest component-state membership function relative to other flash mix components.

If the capacity of the PAC transfer pump is increased, the area of the component-state membership function will increase. This will lead to a direct improvement of the overall system performance. Table 3 summarizes the fuzzy performance indices for both cases (i.e., before and after changing the PAC transfer pump's component-state membership function value). The combined reliability-vulnerability index has increased from 0.699 to 0.988, which means an increase of 41% of the original value. On the other hand, the fuzzy robustness index has increased from -2.120 to -1.127 indicating an improvement of the system robustness.



Figure (21) System-state membership function change for different system components.

Table (3) System fuzzy performance indices change due to the improvement of PAC transfer pump capacity.

Fuzzy performance index	Before change	After change		

Combined Reliability-Vulnerability	0.699	0.988
Robustness (level 2 – level 1)	NA [*]	NA [*]
Robustness (level 3 – level 1)	-2.120	-1.127
Robustness (level 3 – level 2)	-2.120	-1.127

NA^{*} Not-available value as there is no change in overlap area.

Table 4 shows three different changes in the maximum capacity of the PAC transfer pump and their impact on the system fuzzy performance indices. A 5% increase in the maximum capacity of the PAC transfer pump resulted in a more than 7% increase in the combined reliability-vulnerability index with almost no significant increase in the robustness index.

Change in the maximum capacity of the critical component and consequently its membership function results in the appearance of new critical components that control the overall system performance. Therefore, the optimum improvement of system performance can be achieved by an iterative procedure for analysis of the system fuzzy performance indices. Table (4) Change in the system fuzzy performance indices due to change in the maximum capacity of the PAC transfer pump.

	Percentage change of the maximum capacity				
Fuzzy performance index	300%	20%	5%		
Reliability-Vulnerability	0.988	0.921	0.749		
Robustness (level 2 – level 1)	NA [*]	NA [*]	NA [*]		
Robustness (level 3 – level 1)	-1.127	-1.607	-2.100		
Robustness (level 3 – level 2)	-1.127	-1.607	-2.100		

NA^{*} Not-available value as there is no change in overlap area.

5 ANALYSIS OF THE ELGIN AREA SYSTEM

5.1 EAPWSS system representation and data

The system representation provides the integrated layout that reflects the failure-driven relationship among different components. Figure 22 shows EAPWSS with all major components combined in an integrated layout. Component-state and component-failure membership functions are constructed based on the data from the (Earth Tech Canada Inc.,2000), (Earth Tech Canada Inc.,2001), (American Water Services Canada-AWSC, 2003a), (American Water Services Canada-AWSC, 2003a), (American Water Services Canada-AWSC, 2003b), and (DeSousa and Simonovic, 2003). Appendix (I) includes all the input data used in the calculation of the triangular and trapezoidal membership functions representing component-state and component-failure.

5.2 Results

5.2.1 Assessment of the fuzzy performance Indic ices

The same acceptable levels of performance are used in the assessment process (i.e., (0.6,0.7,5.0,5.0), (0.6,1.2,5.0,5.0), and (0.6,5.0,5.0,5.0)). The combined reliability-vulnerability index for EAPWSS is 0.042. This value is extremely low, taking into account that the system is only 4% compatible with the highest level of performance. The fuzzy robustness index for the system is 1.347. Taking into consideration, that this value is the inverse of change in the overlap area, as defined in Equation 18, EAPWSS has low robustness as the overlap area is reduced by more than 74%.

The fuzzy resiliency index value for the EAPWSS is 0.054, which means that it takes the system more than 18 days to return to the full operation mode, as defined by Equation 19. This value is relatively low as it means the system service is disrupted for less than 3 weeks.

5.2.2 Importance of different membership function shapes

As performed in LHPWSS analysis, the effect of the system-state membership function shape is investigated using triangular and trapezoidal shapes. Table 5 shows values of fuzzy performance indices for EAPWSS system.



Figure (22) EAPWSS system integrated layout- Part 1.



Figure (22) EAPWSS system integrated layout- Part 2

Fuzzy performance index	Triangular MF	Trapezoidal MF
Combined Reliability-Vulnerability	0.042	0.017
Robustness (level 2 – level 1)	1.347	3.314
Robustness (level 3 – level 1)	NA^*	NA^*
Robustness (level 3 – level 2)	-1.347	-3.314
Resiliency	0.054	0.054

Table (5) The EAPWSS system fuzzy performance indices for different membership function shapes.

NA^{*} Not-available value as there is no change in overlap area.

As shown in Table 5, the reliability-vulnerability index value has decreased from 0.042 for the triangular shape to 0.017 for the trapezoidal shape (i.e. more than 50 % decrease of the value for the triangular shape). This is similar to the behavior for LHPWSS system as shown in Figure 23. The robustness index values, also, changes with different shapes of membership functions.

For the triangular shape, the robustness index value is 1.347, while it is 3.314 for the trapezoidal shape. It has to be noted that the sign of the fuzzy robustness index indicates the type of change in the overlap area with the corresponding acceptable levels of performance. Therefore, it is more important to observe the absolute value of the fuzzy robustness index rather than its sign.



Figure (23) Resulting EAPWSS system-state membership functions for triangular and trapezoidal input membership functions.

5.2.3 Significance of system components

The change of the system-state membership function is observed to identify the critical system components. Triangular membership functions are used and the resulting system-state membership function progress is shown in Figure 24. Figure 24 shows that the system-state membership function significantly changes twice, after including the PAC storage and after including the PAC metering pump. Similar to LHPWSS system, this is the point where the flash mix system is introduced into the system. Therefore, improvement of the performance of these components will result in the improvement of the overall system performance.



Figure (24) System-state membership function change with introduction of system components.

The PAC components have almost similar component-state membership functions (i.e. for the triangular shape they are (0.00,0.50,1.00)). As a result, the change of maximum capacity of all PAC components is mandatory to significantly change the system-state membership function and consequently the system fuzzy performance indices.

Increasing the maximum capacity of the PAC system to cause a change of the componentstate membership functions by 20% is used to investigate the effect of the change on the fuzzy performance indices. This change will be applied to the modal and the end values of the membership function (i.e., the component-state membership function will be (0.00,0.60,1.20)). Table 6 summarizes the fuzzy performance indices for both cases (i.e., before and after changing the PAC component-state membership function value).

Table	(6)	System	fuzzy	performance	indices	change	due	to	the	change	in	the	PAC
maxin	num	capacity.											

Fuzzy performance index	Before change	After change		
Combined Reliability-Vulnerability	0.042	0.047		
Robustness (level 1 – level 2)	-1.347	-1.210		
Robustness (level 1 – level 3)	NA [*]	NA [*]		
Robustness (level 2 – level 3)	1.347	1.210		

NA^{*} Not-available value as there is no change in overlap area.

The combined reliability-vulnerability index increased by only 12 % (i.e., from 0.042 to 0.047), while the robustness index decreased by 10%. Changing the critical component maximum capacity results in the appearance of new critical components that control the system performance.

6 CONCLUSIONS

The combined fuzzy reliability-vulnerability index, robustness index, and resiliency index are used to asses the performance of the Lake Huron Primary Water Supply System (LHPWSS) and the Elgin Area Primary Water Supply system (EAPWSS). Triangular and trapezoidal membership function shapes are used to examine the sensitivity of these performance indices. They are calculated for arbitrary selected acceptable levels of performance. Three different views of decision-makers are assumed and referred to as reliable, neutral and unreliable levels of performance. The same levels of performance are used for both LHPWSS and EAPWSS systems to facilitate the comparison of the fuzzy performance indices.

Figures 25 (a) and (b) show the three fuzzy performance indices for the two systems. It can be concluded that LHPWSS system is more reliable and less vulnerable than EAPWSS system. The combined reliability-vulnerability index for the LHPWSS system is higher than that of the EAPWSS system for the both triangular and trapezoidal shapes of membership functions (i.e. at least 10 times higher). This is supported by the fact that increasing the system redundancy, by adding parallel and standby components, increases the capacity of the overall system. The LHPWSS system has more than 20 parallel groups and 7 redundant components, while the EAPWSS system has less than 16 parallel groups and 4 redundant elements. This increases the reliability of the LHPWSS system over that of the EAPWSS.



Figure (25a) Fuzzy performance indices for the LHPWSSS and EAPWSS systems for the triangular membership function shape.



Figure (25b) Fuzzy performance indices for the LHPWSSS and EAPWSS systems for the trapezoidal membership function shape.

Additionally, the components of the LHPWSS system are designed with an overload capacity of 35% that positively affects the reliability of the system. As a general conclusion, the LHPWSS system is more reliable and less vulnerable to disruption in service than the EAPWSS system.

Robustness index value shows similar behavior for the triangular membership function shape. The difference in the robustness index values between the two systems is not as high as the difference in the combined reliability-vulnerability index. LHPWSS is more robust than EAPWSS for the two used shapes of the membership function. Therefore, EAPWSS system is more sensitive to any possible change in demand conditions than LHPWSS system as evident form the values of the robustness index of both systems.

The combined reliability-vulnerability index is highly sensitive to the shape of the membership function. Changing the membership function shape from the triangular to the trapezoidal, in both systems, results in a significant decrease in reliability. As an example in EAPWSS, the value of the combined reliability-vulnerability index decreases from 0.042 to 0.017 for trapezoidal shape. In case of robustness index, the change in the value is not as significant as in the case of the combined reliability-vulnerability index.

The recovery time for EAPWSS system components does not exceed 30 days. Some of the components in the LHPWSS system have a recovery time of more than 120 days. Therefore, the fuzzy resiliency index for the EAPWSS system is 4 times higher than for the LHPWSS system. However, the resiliency index is not sensitive to the shape of the

membership function. This is due to the fact that the resiliency index value uses the center of gravity (COG) of the system-failure membership function, and the change in shapes does not affect the value of the COG and consequently the index value.

The developed calculation script can be used to identify critical system components. For example, the PAC components are found to be the critical components for both systems. Slight changes in their maximum capacity significantly affect system performance indices.
7 **REFERENCE**

American Water Services Canada-AWSC (2003a), Elgin Area Water Treatment Plant 2003 Compliance Report. Technical report, Joint Board of Management for the Elgin Area Primary Water Supply System, London, Ontario, Canada.

(http://www.watersupply.london.ca/Compliance_Reports/Elgin_Area_2003_Compliance_Report.pdf)

(accessed November, 2004)

American Water Services Canada-AWSC (2003b), Lake Huron Water Treatment Plant 2003 Compliance Report. Technical report, Joint Board of Management for the Lake Huron Primary Water Supply System, London, Ontario, Canada.

(http://www.watersupply.london.ca/Compliance_Reports/Huron_2003_Compliance_Report. pdf)

(accessed November, 2004)

Ang, H-S and H. Tang (1984), *Probability Concepts in Engineering Planning and Design*, John Wiley & Sons, Inc, USA.

DeSousa, L. and S. Simonovic (2003). Risk Assessment Study: Lake Huron and Elgin Primary Water Supply Systems. Technical report, University of Western Ontario, Facility for Intelligent Decision Support, London, Ontario, Canada. El-Baroudy, I. and S. Simonovic (2003), New Fuzzy Performance Indices for Reliability Analysis of Water Supply Systems. Technical report, University of Western Ontario, Facility for Intelligent Decision Support, London, Ontario, Canada.

(http://www.engga.uwo.ca/research/iclr/fids/Documents/Fuzzy_Preformance_Indices.pdf) (accessed November, 2004)

El-Baroudy, I. and S. Simonovic (2004), Fuzzy criteria for the evaluation of water resource systems performance. *Water Resource Research*, Vol. 40, No. 10.

Earth Tech Canada Inc. (2000), Engineers' Report: Elgin Area Primary Water Supply System. Earth Tech Canada Inc., London, Ontario, Canada.

Earth Tech Canada Inc. (2001), Engineers' Report: Lake Huron Primary Water Supply system. Earth Tech Canada Inc., London, Ontario, Canada.

Ganoulis, J. G. (1994), *Engineering Risk Analysis of Water Pollution: Probabilities*, VCH, Weinheim, The Netherlands.

Hashimoto, T., J. R. Stedinger and D. P. Loucks (1982a), "Reliability, Resiliency, and Vulnerability Criteria for Water Resources System Performance Evaluation", *Water Resources Research*, Vol. 18, No. 1, pp. 14-20.

Hashimoto, T., D. P. Loucks, and J. R. Stedinger (1982b), "Robustness of Water Resources Systems", *Water Resources Research*, Vol. 18, No. 1, pp. 21-26.

Kaufmann, A. and M. Gupta (1985), *Introduction to Fuzzy Arithmetic: Theory and Applications*, Van Nostrand Reinhold Company Inc, New York, USA.

Simonovic, S. P. (1997), "Risk in sustainable Water Resources Management", *Sustainability of Water Resources Under Increasing Uncertainty*, Proceedings of the Rabat Symposium S1, IAHS Publication, No. 240, pp.3-17.

APPENDICES

APPENDIX I: INPUT DATA

I-A LAKE HURON PRIMARY WATER SUPPLY SYSTEM

(LHPWSS)

	-	1000		Capacity		Requirment			Normalized Capacity (TRI)			Normali	zedRequin	Normalized Capacity(Trap)				
	System	Units	Avarage Debuildere	Dateon	Maximum	Yearly Mit	Yearly Zvg	Yearly Mon	Average Daily Par	Decign	Maximum	Yearly Min	Yearty Avg	Valenty Adam	Average Colly Ree	Capacity	Capacity	Masthum
fetalle Crith	οĒ	MID	157.3	340.0	454.5	62.0	197.9	3927	1.5	0.75	1.00	0.12	0.35	0.4	0.8	0.55	D.ET	1.00
Chicolanto J	首員	KAN	362.0	630.0	900.D	215	72.0	130.0	11.40	0.70	1.00	0.00	0.00	11.14	0.0	145	DIK	1.00
RC lateke Pipe	50	MO	192	340.0	404.0	53.0	87.3	38.7	0.35	0.75	1.00	0.12	0.15	0.58	0.35	a ea	D.BT	1.00
Environment Sciences		WLD.	167.3	340.0	404.5	53 D	107.3	255.7	0.35	0.75	1.00	(£12)	0.25	0.56	0.35	0.55	1.67	1.00
Paraping Wells		MLD.	16/3	340.0	454.5	뒷문	87.3	-256.7	0.25	0.75	1.00	632	0.5	0.66	0.35	055	0.67	1.00
Single Speni Pase 1	51	MUB	49.9	76.0	100.0	58	498	812	0.50	0.75	100	637	0.49	0.91	0.50	0.50	0.65	1:00
Variable Spind Parap	친물	MLD	37.4	16.2	115.5	173	40.0	22.4	0.50	0.75	1.00	g (17	0.43	10.0	0:50	0.02	5.67	1.00
Single Speed Pares 7 (Park and	86	M.D.	49.9	750	100.9	16.8	40.0	B1.2	0.50	0.75	1.00	0.12	0.49	0.01	0.40	0.62	0.67	1.00
Versitive Spread Posts (Deck-an)	CHC.	41.0	374	862	195.0	10.0:	290	- D) 2 - 00 4	0.90	875	1.00	0.47	0.43	Digit.	0.50	0.62	DEF	1.00
Single Speed Pamp 2 (Bechap)		M.D	611	75.0	100.0	18.6	19.0	812	0.50	0.79	3.00	0.12	0.59	0.01	0.50	0.62	ner	1.00
PMC Storage Pank		11	0.0	750	150.0	0.1	0.2	0.5	0.00	0.50	1.00	0.00	0.10	0.00	DCD	0.2	DUB	100
PNC Strept Fank PDC Fancter Power		Kalif	80.5	1600	300.0	200	7 4947	6.653	0.30	0.50	1.00	0.76	1.21	2.22	0.20	8.45	BSD	1.00
PAC Taxoder Pump		KgN	0.00	100.0	300 D	327.0	419.7	057.1	0.20	0.00	1.00	a.m.	1.41	2.22	0.20	CI 40	020	1.00
Alaro Storage Task		n (0.0	35.0	68.0	1.1	1 3 <u>9</u> 0	7.0	0.00	0.50	1.00	0.00	0.25	0.11	0.00	0.25	5.75	1.00
Adam Sharege Lank		10	33	44	80.5	15	240 L	70	0.50	0.90	1.00	0.00	0.65	1.05	0.50	0.25	0.46	1.00
ALAND Fransfer Forap	E.	n ² 31	33	48	87	F. 12	31	70	0.50	0.73	1.00	0.28	0.45	1.05	0.50	0.65	0.65	100
Pluch Mix Col/1-T	5	MA	707	1/00	217.5	20.5	77.3	127.9	0.35	0.75	120	6.12	0.34	0.55	0.25	0.65	1.67	1.00
Phane and Calific Cali	8	ULD.	00	5.6	120	00	197	6.9	0.05	0.50	1.00	0.12	0.14	0.96	0.00	0.55	0.57	100
Polymer Storage Touri	2	102	0.0	6.6	136	00	4.4	8.8	100	0.50	1.00	0.01	0.33	0.65	0.00	0.75	875	1.00
Pulyoser Tousder Parop	6	Lu.	0.0	32	64	00	2.1	4.2	0.00	0.60	1.00	0.08	0.33	0.025	0.00	0.25	075	1.00
Polyner Taxabir Parip	2	생	0.0	32	24	00	135 B	0.02	0.00	0.40	1.00	0.00	0.15	0.65	0.00	<u>A</u> D	0.05	100
Polymer Mix Tash		10	0.0	1 34	6.0	0.0	22	4.4	0.00	0.60	1.00	0.0	0.73	0.65	0.00	13	8.76	1.00
Polynoir Feel Party		18	01	0.4	0.8	1,0	0.2	0.4	0.97	0.00	1.00	0.41	0.38	0.65	D.(7	0.30	D370	1.00
Polymer Past Parcy		La	111	120/0	0.6	20	8 22.9	0.4	0.10	0.55	1.00	0.17	0.31	0.65	0,17	0.38	0.79	1.00
Flick Blin Call 92		M.D.	78.7	1700	227.5	36	77.2	127.9	0.35	0.75	1.00	613	0.14	0.55	0.95	0.55	0.67	100
PWC Traceler Planys		Kgʻti	60.0	160.0	300.0	2011	497	923	0.20	0.60	1.00	0.78	1.4]	222	0.70	0.45	080	1.00
Celler	1.00	91.0	55	20.0	42.6	0.0	10.9	- 10.0	0.00	0.49	1.00	0.11	0.45	0.75	0.10	0.99	0.80	1.00
640.64	8.0	W.D.	RO	26.8	125	6.6	18.9	31.0	-0.19	0.50	100	0.04	0.65	8.75	RIR	0.39	DAD	100
6208-2	목표	91.0	Ð.D.	25.9	42.5	66	193	- 39.0	0.19	0.69	1.00	0.18	0.45	0.75	0.19	0.38	080	1.00
Cedari	88	병왕기	50	28.2	414	66	193	33.0	0.12	0.69	170	0.3E	0.85	0.75	0.19	0.39	0.00	1.00
CAURA	8	MO	80	22	120	66	識	20.0	0.00	0.50	1.00	0.16	0.45	0.05	0.19	0.39	0.80	100
Cell N-2		MLD	BD-	201	- 25	66	19.3	- 32.0	0.58	DSH	1.00	à 14	0.45	0.75	010	0.3	880	1.00
Seating Task H	255	48.0	78.7	100	221 3	(0,3	30.5	63.9	0.36	0.05	1,00	0.06	0.17	0.20	0.35	口结	0.07	1.00
Setting Tank 1-3	***	95.0	78.7	1900	302.5	113	30.0	61.9	0.35	0.75	1.00	0.06	0.12	0.26	0.25	0.55	Det	1.00
Setsling Tank 6-7	SEG	MLD	787	170.0	227.3	193	385	61.9	0.35	0.75	1.00	0.05	0.17	0.28	0.35	0.5	0.67	1.00
P May 53		MLB	121	26.3	37.9	14	12.9	213	0.35	A 15	1.00	4.12	9.14	0.60	0.35	0.55	0.07	1.00
FShar 52 Filter 8.7		MLD (28.3	474	1 11	129	213	0.25	0.75	1.00	0.12 H	0.24	0.55	0.35	0.50	8.67	1.00
Ether #2		M.D	13.1	28.3	37.9	11	129	21.3	0.95	0.75	1.00	0.12	0.14	0.66	0.15	0.65	0.67	1.00
Filter#1	E.	MO	1 101	28.7	37.9	2.4	129	21.3	0.36	0.75	1.00	0.12	0.34	0.96	0.35 -	0.份	0.67	1.00
1 West #-2	i i i i i i i i i i i i i i i i i i i	MLO	-01	20.7	37.9	1.11	129	213	0.35	0.75	100	0.12	0.15	0.50	0.35	0.55	0.07	1.00
Film 11-2	8	MLD.	121	28.3	37.9	13	(2.9	213	0.35	0.75	1.00	615	0.34	0.56	0.35	0.55	0.67	1.00
Filter V-2	5	MO	1.121	28.7	39.9	44	128	213	0.26	0.75	1.00	0.12	0.34	0.65	0.35	0.95	0.67	1.00
Pather U-1	÷.	M.D.	-111	20,3	37.2	64	13.9	215	1.36	0.75	1.00	C (3)	014	0.60	0.00	0.10	1.07	1,00
Filter W/2	8	MD		26.3	329	1.12	129	213	0.25	0.55	120	6.32	0.34	0.56	0.35	0.55	0.67	1.00
Celoninator N		Kalit	360.0	630.0	300.D	34.0	545	91.0	0.40	0.70	1.00	0.04	0.36	0.10	0.40	0.65	D.85	1.00
Chlorikator N		Kati	380.0	630.0	1005 E	340	63.5	R1D	0.45	0.70	1.00	0.01	0.85	010	0.0	0.55	0.05	1.00
Citar Well 2		Ma	124	1059	1994	12	129	21.5	1193	0.66	100	0.01	0.13	0.03	0.33	0.50	D83	100
Cikar Well 3		WD	42.4	105.9	19.4	1.11.	129		0.23	0.85	1.00	0.00	0.19	013	0.32	0.00	D.82	1.00
Phone 1		MO	787	0.4	1001	25.2	76.9	127.9	0.79	0.99	1.00	10.21	0.75	1.21	0.79	0.54	0.96	1.00
Parap 2 Parap 3	252	MO	787	10.4	1001	23	75-7	127.9	100	Data	1.00	0.25	0.76	1.20	0.09	0.54	0.96	1.00
Parap #(Back-ap)	128	MLD .	78.7	EB.4	100.1	33	75.9	127.9	0.79	0.99	1.00	0.25	0.76	1.29	0.79	0.64	0.95	1.00
Plymp-2 (Birch-and		MLD	19.7	10.4	300.1	2.3	759	127.9	0.78	0.09	1.00	0.34-1	0.35	1.29	0.70	0.62	0.96	1.00
	2 -	- and the second	157.3	152.3	277 2	353	75.9	(27.9	0.69	0.99	100	641	0.13	0.66	0.69	0.77	DIO	101
Alive 1	1 Aug	10.0	1000	Contraction of the	1447.08	0.000	Contes	201809	1.0000	ACCES 1		80.5	2000	alder ,	2002-00	Cace.	Come.	10000
21.26240)	ave.	10000	1 6655	1 careto	1/1/25	1.188.00	12305	Yester 1	10.285	0.0000	17282	2400	33220	199	5630	123/25	1,222	1980
Mar. 2	^o	WR	197.0	192.1	207.2	-63	75.9	127.9	0.69	DOBIN	1.00	0.11	0.11	0.66	0.69	0.77	0.92	4.00
Peop 1		M O	15	103	101	16	62.3	1001	0.05	0.62	100	0.05	0.97	100	006	8.8	0.76	100
Picop 2	u gu	MED	3.1	52.3	100.1	25	62.3	100.1	0.05	0.53	1.00	0.05	0.17	4.02	DÓS	0.25	0.76	1.00
Phone 3	Ash ost	MLD	4.8	52.3	100 1	14	52.5	IDD;1	0.05	0.52	1.00	0.05	0.12	100	BUS	0.25	2.76	1 00
Photo 4 Reservoir	86	MLD .	4.6	52.3	187	48	52.3	1001	085	0.42	1.00	0.06	0.92	100	0.05	0.06	D.76	100
CHEE		Mit	00	157	27.2	00	0.6	15.5	800	0.50	1.00	0.00	014	0.48	0.00	0.0	D.PS	1.00
640.2	295	W	DB	137	22.2	0.0	-CE	133	00.0	0.50	1.00	0.01	0.34	0.49	0.00	0.25	0.75	1.00
Cell 1	East	MUD	0.0	18.7	27.3	0.0	86	13.3	0.00	0.50	1.00	0.01	0.24	0.46	0.00	0.25	8.75	1.00
Gell 4	200	MUD	0.0	137	23	00	8.6	13.9	0.00	0.60	1.00	0.04	0.74	0.48	000	0.35	0.85	1.00

	COLORISON OF	112020	Normalized Requirment(Trap)			Element-State (Tri-MoS)			Element-State (Trap-MoS)				Elem	ent-Failu	re (Tri)	Element-Failure (Trap)				
	System	Units	Anarogo	Capacity	Capacity	Maximum	-	b	o	4	Ď.	-2	e	-	b.	0	4	ð	4	ď
fetalle Crip	οĒ	MLD	0.12	0.46	0.45	0.56	0.22	0.40	0.69	4.22	0.09	(1.0)	0.69	0.0e	0.13	0.17	0.00	0.11	0.14	0.17
Chiceleator /	19	KgN	10.00	11.11	11.11	014	0.35	0.62	0.07	0.76	0.64	12.74	0.97	0.01	10.0	apr	bot	ua)	DO1	nai
RC lateke Pipe	=6	MLO	0.12	0.45	0.45	0.95	4.12	0.40	0.00	-9.22	0.09	0.41	1165	0.05	p15	0.17	0.36	U II	0.14	27
Environment Service	11010	MLD .	0.52	0.40	0.45	0.50	0.22	0.50	6.00	0.22	0.09	0.41	0.89	0.04:	0.05	1.00	DDA 000	0.22	0.35	0.50
Chlormator #	2.	Kali	0.08	0.it	0.11	0.14	0.25	0.62	0.97	0.26	0.44	0.74	0.97	0.01	0.01	0.01	D.D1	0.01	0.01	8.01
Silvade Speed Rusep T	63	MU	71.0	0.65	0.65	19,0	0.31	り渡	0.00	4.34	-0.03	0.22	0.83	0.02	261	東田	20.02	1.98	3.34	5.00
Simple Speed Pyrop	A.	MD	0.12	8.90	0.64	Dát	431	0.32	0.03	431	-803	0.32	680	0.02	2.51	5.00	0.82	168	-394	5.00
Single Speed Panys T (Bach-up)	2"	MLD	0.17	0.66	0.66	0.81	-0.31	0.26	0.83	41.31	0.03	0.22	0.03	0.02	2.51	5.00	0.02	1.68	334	6,00
Veriable Speed Puop (Back-up) Strate Scenet Pares 7 (Back-up)		MD	0.37	0.95	0.62	0.91	0.11	250	0.00	131	10.0	1.32	083	0.02	201	5.00	200	169	334	500
/MC Savuga Fank		m'	00.0	0.00	0.00	DIID	au	0.5p	1.00	0.00	0.25	0.75	100	0.04	3.62	10.7	0.04	2.30	4.65	7.00
PAC Shavpe Tank		10	6.00	0.05	0.00	9,00	6.00	0.50	100	0.00	0.25	075	100	0.04	342	7:00	0.04	236	410	7.00
PhC Tavoder Puop		Kald	0.76	1.70	1.01	2 22	2.02	-0.00	0.24	-212	-Car	-0.02	0.24	0.04	3.62	7.00	0.04	2.30	4.20	7.00
Adam Storage Fask			0.05	EX2E-	0.00	0.01	40.11	0.45	0.57	4.13	Q.YP	080	0.97	0.04	- 342	1.00	DIR	2.95	4.08	7.00
Adam Science Lank Adam Transfer Paren		10.74	0.05	0.00	0.05	1.05	3.65	0.45	0.75	2.07	D14	0.62	0.75	0.04	362	7.00	0.04	2.35	4.50	7.00
Alano Pransfer Parap	5	m ¹ M	0.25	0.57	0.76	3.05	0.66	0.26	0.75	司历	014	0.29	0.75	0.04	362	7.00.	0.04	2.36	4.68	7.00
Pluck Rite Califit Flock Mir Califit	5	MLD	0.12	0.48	0.45	0.58	-0.22	0.40	0.55	422	0.30	0.42	0.85	0.50	7.25	14.00	0.50	500	9.50	14 00
Polyner Storage Tosh	8	10	0.00	0.49	0.49	0.65	-0.95	0.18	1.00	4366	0.21	0.26	1.00	0.50	7,25	14,00	0.50	5.00	3.50	14-00
Polyoner Storage Tous	2 E	10	0.00	0.42	0.49	0.65	0.65	0.10	1.00	1.65	0.34	0.26	1.00	0.60	7.8	34.00	0.90	500	950	14.00
Polymer Transfer Parap	1 2	Lix	0.00	0.42	0.40	0.65	-0.65	0.10	100	3.65	-0.34	0.25	1.00	0.50	128	14.00	0.50	5.00	9.50	14.00
Potymer Mix Tank		10	0.00	0.49	0.49	0.65	-0.05	0.10	1.06	4.66	324	0.26	1.00	0.50	1,25	14:00	0.50	600	950	10(10)
Polyner Mor Lank Relevant Faat Polyn		10	0.00	0.49	0.49	0.65 D.05	20.99	0.18	0.02	-0.00	0.54	0.25	0.00	0.50	120	34.00	0.50	5.90	950	14.00
Polymer Paul Pump		La	0.11	9.61	0.51	0.65	-0.48	0.00	0.50	41.48	0.14	0.28	0.89	U 5D	1.25	14.00	0.50	500	0.50	14,00
Flink Mit Cel/9-1		MLD	0.12	0.45	0.45	0.56	-0.22	0.42	0.99	4.22	0.10	0.42	0.09	0.50	7.05	14.00	0.50	500	950	14.00
PWC Treveler Plung		Fatt	0.76	172	181	272	207	0.80	0.24	210	141	8.00	0.34	0.04	1 362	7.00	0.04	236	4.68	200
Cell AT		MLD	0.75	080	0.60	0.16	0.40	0.14	9.54	4.56	0.0	2.15	160	0.00	0.60	1.00	0.00	0.01	0.67	1.00
Ce0.61	a a	W.D	0.16	0.60	0.60	0.75	-0.96	0.14	0.64	16	0.21	0.15	0.84	0.00	0.50	1.00	-500	0.93	0.67	1.00
CALLAR	88	91.0	0.15	0.90	0.60	0.75	0.96	0.14	8.94	角筋	0.21	业绩	0.84	0.00	0.60	1.00	0.00	0.33	0.62	NO
Cell B-F	28	M.C.	0.15 0.7K	0.00	040	0.75	-0.50	0.14	0.04	-0.50	-0.21	0.10	085	0.00	0.00	1.00	0,00	0.33	DET	1,00
Cell 8-1	- E C	MLO	0.16	0.90	0.60	0.75	0.99	6.14	0.64	0.56	-0.21	0.15	0.84	6.00	0.50	1,00	0.00	0.99	0.67	1.00
Cell N-2		MLD	0.45	881	0.60	875	-0.96	1514	0.54	0.00	021	0.10	0.94	0.00	0.90	1.00	0.00	0.33	0.67	1.00
Setting Tank 1-2	1 2 2 3	95.0	0.08	0.25	6.25	0.20	u m	0.55	12.54	2.08	11.02	0.65	1.94	0.08	0.54	1.00	0.05	0.22	0.60	1.00
Settling Tank Srt	De la compañía de la comp	M.D	0.06	0.29	0.23	8.28	0.06	0.98	11.54	0.06	0.32	0.65	0.94	0.00	0.54	100	0.08	0.29	0.69	1.00
Film 13		MLD	0.12	0.6	0.45	0.55	0.32	Dat	0.00	4.22	0.10	0.42	0.00	0.04	500	10.00	0.04	136	6.00	100
P Shar 52		M.D	0.12	0.45	0.45	0.66	411	£1.41	0.95	4.72	0.92	0.42	085	0.04	5.02	10.00	1129	3.30	2.68	10.00
Film #3 Film #2		MD	0.32	0.45	0.45	0.58	0.22	0.41	0.08	41.32	0.90	0.42	049	0.04	502	10.00	-0.54	3.35	6.6B 6.6B	10-00-
Filter #-1	12	MO	0.32	0.45	8.46	0.95	0.22	0.45	1.00	41.28	0.30	842	0.98	0.04	5.02	10.00	0.04	3新	5-68	10.00
Filter #-2	E E	MO	0.12	0.85	0.45	0.55	-0.22	0.61	0.02	4.22	0.92	0.42	0.005	0.06	5.02	92.00	0.02	136	1.60	10 00
Film N/2	8	MLD.	0.2	0.45	0.45	0.58	0.22	0.41	0.86	0.22	0.10	0.42	0.00	0.04	8.02	12.00	0.04	3.36	5.68	10.00
Filter V-2	6	MO	21.0	0.45	0.45	8.66	-0.22	Del	0.18-	0.22	0.10	0.42	0.09	0.94	5.02	\$2.00	0.04	3.36	6.6B	10.00
Chier U-1	1 12	MLD	0.12	1.45	0.45	0.95	-0.22	D.41	0.00	41,72	0.90	6.42	100	6.04	5.02	10,00	DIM	130	5.68	10 00
Filter W/2	1	M.D	0.12	0.45	0.46	0.46	-0.22	0.41	0.00	-0.22	0.0	0.42	0,08	0.04	5.82	\$7.00.	0.04	930	6.68	10.00
Chloringtor N Chloringtor N		Kalit	0.04	0.02 p.gr	0.08	0.10	0.30	0.84	0.96	0.30	0.47	0.78	0.96	10.0	0.01	0.01	D.01	0.01	0.01	0.01
Citur Well 1		MLD	0.05	U.it.	11.11	Dis	0.20	0.56	U.W.	0.20	0.29	215	0.97	0.00	8.50	1.00	0.00	0.53	0.67	1.00
Clear Well 2 Clear Well 3		MB	0.09	0.11	0.11	0.19	0.20-	0.58	0.57	0.20	0.99	0.79	0.97	0.00	0.50	1.00	200	0.33	067	1.00
Pyop 1		MO	0.25	1.01	1.02	1.28	0.49	Coda	0.74	41.49	0.10	-6.07	076	0.04	60.02	120.00	0.04	40.08	10.00	120.00
Purap-2	622	91.0	0.25	101	102	1.28	-0.49	0.13	0.75	-0.45	0.10	-0.07	自市	0.04	1 00.02	120.00	0.04	40.00	10.05	120.00
Phone 3 Phone 1 (Back and	115	MO	122	101	100	1.20	1.0	0.13	0.75	-0.49	-0.18	407	0.75	0.04	60.02	125.00	0.04	40.00	50.01	120:00
Pavap 2 (Back-ap)		MLD	0.8	101	/182	1.29	-0.49	0,13	0.25	0.49	0.10	0.07	876	0.84	1 60.02	131.00	0.54	47-10	1000	120.00
	ê -	10000000	0.11	0.5	0.45	p.co	0.02	per	0.00	042	10.00	0.48	nim	0.00	191	3.00	000	107	201	707
Allow 1	1.5	10.0	1.400	9992-1	19.45	0.99	6.22	-1489	9.05	1018	age)	CUMP:	0.08	10.00	1.99	3365	2.6105	201.	-3600);	200
10040	ave a	10000	1 3325	1 100	10000	1 25	49633	12000	8576	12333	6.5385	10000	10000	19723	10.33	8825	166	- 22577	1833	1 - 22.1
Main 2	ß	. W.B.	0.94	U.M.	0.45	0.56	0.0	100	0.94	0.13	0.32	11.48	609	0.02	1.51	3.01	0.05	1.01	Stu.	300
Parap 1		M.O	0.06	0.76	0.76	100	-0.95	0.00	0.95	0.96	-0.49	0.00	0.96	0.08	60.04	120.00	0.08	41.06	60.0G	12000
Fixap 2	ten u	68.0	0.05	8.70	878	1.01	-0.95	0,00	0.50	42.96	0.01	00.0	295	12.02	10.01	120.00	0.08	43.00	10.03	120.00
Parate 4	000	MLD	0.05	0.76	0.76	1.00	4.95	0.80	0.95	4196	0.40	0.00	0.95	0.08	F 60.04	120.00	0.00	43.06	80.03	120.00
Reservoir	10 °	M.D	1 000	0.75	0.75	100	-1.00	0.00	1.00	-100	-0.90	0.00	100	0.08	t enda	120.00	0.08	41.05	50.08	12000
Cell 2	2 A F	MLD.	0.00	0.32	8.1	0.45	-0.48	DUZB	1.00	-12.425	011	0.30	1.00	0.00	0.64	1.00	21.08	0.39	980	1.00
Ce0.2	E S S	MLD	0.00	0.96	9,30	0.48	지석	0.26	100	-0.48	-0.11	0.38	1.90	2.08	R64	1.00	0.00	0.29	0.69	1000
640.5	286	MLD	010	0.25	0.00	0.49	1.48	5.2	1-00	40,48	011	0.30	100	0.06	1 054	1.00	5.00	0.39	020	100
A SAME AN	and the second se		The second se	and the second sec	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	and the second sec	5 St. 1975	10000	the second se	10 M 10 M 10 M		COLUMN TO A STREET	11111	and the second sec	1.1 1700		and the second se	10,000	A DESCRIPTION OF THE OWNER OWNER OF THE OWNER OWNER OF THE OWNER OWNER OWNER OWNER OWNER OWNE OWNER OWNE OWNER OWNE OWNE OWNE OWNE OWNE OWNE OWNE OWNE	The second se

APPENDIX I: INPUT DATA

I-B ELGIN AREA PRIMARY WATER SUPPLY SYSTEM

(EAPWSS)

1.0000000000000000000000000000000000000		EN	eme	nt ID	1				Capac/t	у	8	Requirmen	र्ग	Normal	ized Capac	ity (TRI)	Normali	zedRequin	ment(Trl)	Non	nalized C	apacity(Trap)
System		53(3)	649	Elevie	n	No.	Units	Average Only Pow	Design Caseaty	Moumum Chericald	Veasy IAn	Yeeny Avg	Fearly Max	Average Dely fax	Design Ciepectic	Maximum Orefland	Yearty Ada	Younty Aug	Yearny Max	Avevage Dark for	Canada/	Capacity	Maximum Overland
ananto Coli Sodium Physochiania NC Induke Pipe Plant Drain Juzuke Pipe	Intake Syntem	000	1 1 1	0 0 0		1 2 3 4	MLD MLD MLD MLD	記4 524 位4 414	91.0 91.0 91.0 91.0	181 D 185 D 180 Ú 191 D	85 83 83 83	124 524 524 524 524	71.8 71.8 71.8 71.8	0.39 0.29 0.39 0.39	0.50 0.50 0.50 0.50	1,00 1,00 1,00 1,00	0 19 1 10 0 19 0 19	0.9 0.20 0.95 0.95	0.40 0.40 0.41 0.40	0.29 0.29 0.29 0.29	0.40 0.40 0.40 0.40	0 /6 0 /2 0 /2 0 /2 0 /2	1 00 1.00 1 01 1 01
Towelling Sciences Farvelling Sciences Pringing Work 1 Pringing Work 1 Pringing Work 1 Pringing Pringer Prings 1 High Deckinger Prings 2 Com Discharge Prings 2 Com Discharge Prings 2	Low Lifting System	00000000		0060000000	1700 4 5 6 1 8 9	567551212	MLD MLD MLD MLD MLD MLD MLD MLD	363 367 367 363 363 52 363 52 363 52 434	455 455 455 455 345 709 345 709 345 709	90.5 90.5 90.5 90.5 90.5 97.0 1 27.0 97.0 97.0 97.0 97.0 97.0 97.0 97.0 9	85 85 85 114 42 114 42 42	124 124 124 124 124 125 105 105 105 105 105 105 105 105 105 10	71.6 71.8 71.8 71.5 86 37.3 86 37.3 86 37.3	020 029 029 030 030 030 030	050 050 050 052 052 052 052 052 052	100 100 100 100 100 100 100 100		0.50 0.98 0.58 0.30 0.30 0.30 0.30 0.30 0.30	0.70 0.79 0.79 0.79 0.41 0.41 0.41 0.41	029 029 039 030 030 030 030 030	0.40 0.40 0.40 0.40 0.41 0.41 0.41 0.41	0月 0月 0月 0月 0月 0月 0月 0月 0月 0月 0月 0月 0月 0	1 00 1 00 1 00 1 00 1 00 1 00 1 00 1 00
PHC Stronger 1 PHC Stronger 2 PHC Transfer Parts 3 PHC Transfer Parts 3 PHC Transfer Parts 3 PHC Transfer Parts 3 PHC resoluting Parts 3 Alaw Stronger 1 Alaw Stronger 1 Alaw Stronger 1 Alaw Stronger 1 Alaw Stronger 1 Alaw sentening Parts 3 PHC resoluting Parts 3 PHC resoluting Parts 3 PHC resoluting Parts 3 PHC resoluting Parts 3 Pholyses Birt UphA Polyses Birt UphA Polyses Riv UphA Polyses River Parts 4 Polyses River River Parts 4 Polyses River Parts 4 Polyses River Parts 4 Polyses River River Parts 4 Polyses River River Parts 4 Polyses River Parts 4 Polyses River Parts 4 Parts 4 Polyses River Parts 4 Parts 4 Pa	Flash Mr. System				12345678901234567891112345678	· · · · · · · · · · · · · · · · · · ·	Profiles State	00 00 00 00 00 00 00 00 00 00 00 00 00	855 855 32 35 35 35 55 85 85 85 85 85 85 85 85 8	(710 (710) 63 63 63 63 63 63 63 63 63 63 63 63 63	00 00 00 00 00 00 00 00 00 00 00 00 00	01 01 00 00 00 00 00 00 00 00 00 00 00 0	62 00 00 02 01 01 01 01 01 01 01 01 01 01 025 01 025 038 038 038 030 05 05 05 05 05 05 05 05 05 05 05 05 05	0 000 0 000000	0.60 0.50 0.50 0.50 0.50 0.50 0.50 0.50	1.00 1.00	000 000 000 000 000 000 000 000 000 00	000 000 000 000 000 000 000 000 000 00	000 000 000 000 000 000 000 000 000 00		132 035 035 035 035 035 035 035 035 035 035	4 株 4 株 4 株 4 株 5 株 5 株 5 6 7 6 7 7 7 7 7 7 7 7 7 7 7 7 7	1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00
C+0+1 C+0+1 C+0+2	seculation System	0 0	4	0 0 0	1 2 1	12 1) 44	HLD NLD NLD	00 00 00	455 455 455	91.0 91.0 91.0	лл 88 8 <i>0</i>	11 131 111	17.9 17.9 17.9	0.00 0.00 0.00	0 60 0 50 0 50	1.00 1.00 1.00	0.10 0.10 0.10	0.14 0.14 0.14	0.20 0.20 0.20	a 10 0 00 0 00	425 626 035	0 M 0 M 0 形	1 00 1 05 1 01
Cell #3 Sectomentation Tank I Sectomentation Tank I Pleasable Strenge Z Pleasable Strenge T Pleasable Sectomer Pleasable Sectom	Sedmentation Pk System	000000000000000000000000000000000000000			1 3 3 4 9 6 7	15 10 10 10 10 10 10 10 10 10 10 10 10 10	HAD HAD HAD HAD HAD HAD HAD HAD HAD	0.0 45 45 00 00 00 00 00 00	455 350 260 125,0 45 45 7,4 7,4	91.0 45.4 46.4 250.0 9.0 9.0 9.0 14.6 14.0	6.8 17.6 17.6 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0	13.1 36.2 37.0 37.0 0.0 0.0 25 25	17 B 35 9 35 9 78 0 00 50 50 50 50	0.00 0.10 0.00 0.00 0.00 0.00 0.00 0.00	010 055 013 050 050 050 050 050 050	100 1,00 1,00 1,00 1,00 1,00 1,00	0.00 0.00 0.00 0.00 0.00 0.00 0.00	214 0.58 0.68 0.15 0.00 0.00 0.00 0.00 0.07 0.17	0.30 0.79 0.32 0.00 0.00 0.00 0.34 0.34		0.8 0.8 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5	0.4 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5	1 00 1 00 1 00 1 00 1 00 1 00 1 00 1 00
The I File I File A Cour Well Oursin Research	Fibration System	00000		000000	19466	115679	HLD NLD HLH HL	00 00 00 00	150 150 457 454	30.0 30.0 30.0 30.0 21.5 70.5	5.8 9.0 8.8 35.5 15.5	101 101 101 104 104	17.8 17.9 71.5 71.5	00.0 00.0 00.0 00.0 00.0	010 050 050 050 050	1.00 1.00 1.00 1.00 1.00	0.22 0.25 0.35 0.35	0.44 0.44 0.57 0.58	0.60 0.60 0.60 0.60 0.70	0.00 0.00 0.00 0.00 0.00	035 035 035 035	0.75 0.75 0.75 0.75 0.75	1 00 1 00 1 00 1 00 1 00
Providentiality Provp 3 Provp 7 Provp 4 Provp 2	High Lifting Systems	00000	17777	00000	13346	99 81 82 83	Hari HLD HLD HLD HLD	00 201 281 281 281 201	1250 40.0 40.0 40.0 40.0	250.0 52.0 52.0 52.0 52.0	50 5.5 5.5 8.8 8.9	800 121 131 131 131	17.9 17.9 17.9 17.9	000 054 054 054 054 054	050 0.77 0.77 0.77 0.77 0.77	1.00 1.00 1.00 1.00 1.00	0.04 0.17 0.17 0.17 0.17 0.17	102 0.25 0.25 0.25 0.25 0.25	012 035 034 034 034	6 00 6 54 6 54 6 54 6 54 0 64	0.3 0.65 0.65 0.65 0.65	675 070 039 039 039 039	1.00 1.00 1.00 1.00 1.00
Mar	Conveyance System	0	8	q		. 51		52.4	8210	181.0	362	\$24	71.8	0.29	045	1,90	a.19	0.29	0.40	0.39 /	a #	0 73	't.08
TerminetReservoir 2	Boarting System	0	9	¢ ¢	1	- 15	MLD.	20 00	197	27.5 27.3	nn fui	נט טו	26.2 36.2	0.00 0.00	0:50 0:50	1.00 1.00	a 20 0 00	0,40 0.48	0.96 0.96	a ta 0 00	0.25 0.25	0.75 0.75	r.00 7.00

260.02.55		Elen	nent ID	1.228	10.3523	Norm	alized Re	quimen	(Trap)	Elemen	nt-State (rd-MoS)	Eler	nent-Stat	te (Trap-I	MoS)	Elem	ent-Failui	re (Tri)	E	ement-Fa	ullure (Tra	(ay
System		Sjaten	Elemen	NO.	Units	Average Curs. Ste	Capacty	Canady	Manathan Charleser		2	· · · ·		8	B		14	ii	e				0
intele Criti Sochen Ageocelitatha PC Instite Pipe Plant Davis Intele Pipe	kraine Bystem	0 1	0.000	2	MLD MLD MLD MLD	014 015 019 019	0.94 0.04 0.94 0.94 0.05	034 034 034 034	((4) 0 40 0 41 0 41 0 43	41 41 41 41	821 821 821 821	0.8) 5.81 0.81 0.81	011 011 011 011	0.65 0.85 0.85 0.85	0.42 0.42 0.42 0.42	0.81 0.01 0.01 0.01	0.08 0.04 0.08 0.08	019 342 013 7 013	0.12 T.00 0.17 0.17	0.18 0.14 0.19 0.90	015 236 031 0.11	014 428 014 014	0.17 2.00 0.17 0.12
Termiting Science Termiting Science President Well 1 Anaphing Well 1 Anaphing Well 1 Ana Disclosury Promp 7 High Discherge Promp 2 Cole Discherge Promp 2 Cole Discherge Promp 2 R Januar Pape	LowLifting System	00000000000000000000000000000000000000		5 67 8 9 10 11 12 13	NLD NLD NLD NLD NLD NLD NLD NLD	0.39 0.39 0.39 0.39 0.30 0.30 0.30 0.30	00 00 00 00 00 00 00 00 00 00 00 00 00	0.09 0.09 0.09 0.03 0.35 0.36 0.36 0.36 0.36	039 039 039 041 041 041 041 041 041 041	0.50 0.50 0.50 0.11 0.11 0.11 0.11 0.11	408 400 400 622 622 625 625 625 625 625 625 625 625	861 061 861 060 080 080 080 080	0.00 0.00 0.00 0.00 0.01 0.01 0.01 0.01	029 029 029 029 029 025 025 025 025	0.08 0.00 0.00 0.41 0.41 0.41 0.41 0.41 0.41	587 067 067 069 068 068 068 068 069 069	012 004 004 004 004 004 004 004 004 004	02 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0	0.50 0.55 0.60 0.00 27.00 27.00 27.00 27.00 27.00 27.00 27.00	014 014 014 014 014 014 014 014 014 014	019 019 019 700 700 700 700 700	0.56 0.38 0.36 0.36 14.01 14.01 14.01 14.01 14.01 14.01	050 0.50 0.60 21.00 21.00 21.00 21.00 21.00 21.00
PAC Shaper 1 PAC Shaper 1 PAC Shaper 2 PAC Towards Pares 2 PAC Towards Pares 2 PAC Towards Pares 2 PAC Towards Pares 2 PAC Shareing Pares 3 PAC Shareing Pares 3 Pares 3 Pa	Fisach Mix System			HP#T#PBLZDHABTTBBBLZHASSATBBB	eri UA UA UA UA UA UA UA UA UA UA UA UA UA	0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000	000 000 000 000 000 000 000 000 000 00	0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.0		000 500 200 200 200 200 200 200 201 200 201 200 200	0000 0550 0550 0550 0550 0550 0550 055	100 100 100 100 100 100 100 100 100 100	000 000 000 000 000 000 000 000 000 00	0.52 0.52 0.55 0.55 0.55 0.55 0.55 0.55	でたたたたたたたたたたたたた。 日本の日本の日本の日本の日本の日本の日本の日本の日本の日本の日本の日本の日本の日	100 100 100 100 100 100 100 100 100 0000	000 000 000 000 000 000 000 000 000 00	2.2.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4.4	600 600 600 600 600 600 600 600	0.000 0.180 0.180 0.180 0.180 0.180 0.180 0.180 0.180 0.180 0.180 0.180 0.180 0.180 0.180 0.180 0.180 0.180 0.190 1.000 0.1900	12方2127272272225222233310000000000000000000000000000	2.4 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5	5.00 6.00 6.00 5.00 5.00 5.00 5.00 5.00
GARIA GARRA GARIZ	eoculation System	0 4	0 1 0 2 0 3	42 43 44	MLD MLD MED	010 010 010	0.17 0.17 0.17	0.17 0.17 0.17	0.20 0.20 0.20	41.20 41.20 41.20	0.36 0.36 0.36	0.90 0.90 0.90	-0.20 -0.20 -0.20	0.98 0.99 0.39	0.98 0.58 6.80	0.94 0.94 0.94	0.04 0.04 0.04	3-52 3-52 3-52	7.03 7.00 7.00	0.14 0.14 0.14	2.30 7.36 2.36	4.68 4.68 4.68	7.00 2.00 7.00
Call F3 Sectionate/on Tank I Sectionate/on Tank I Pro-Chartourur Flannin Schwege 2 Flannin Schwege 1 Flannin Fael Pange Flannin Feel Pange (Dach-og)	Setimentation FA	4 000000		45 47 48 49 59 59 51 52	MLD MLD NLD Xigid M ¹ A ¹ LN LN LN	0.10 0.39 0.39 0.00 0.00 0.00 0.00 0.00	6.7 067 0.11 000 000 025 0.25	0.47 0.50 0.90 0.27 0.90 0.90 0.90 0.29 0.29	80 80 80 80 80 80 80 80	420 415 419 432 508 508 508 508 508 508 508 508 508 508	0 2 4 00 4 00 0 37 0 90 0 50 0 30 0 30 0 30	0.90 0.61 0.61 1.00 1.00 1.00 1.00	40.20 4058 4059 40.32 000 600 4034 4034 6034	018 035 035 035 035 035 030 030	0.58 0.10 0.50 0.55 0.75 0.75 0.50 0.50	0.61 0.61 1.08 1.08 1.08 1.08 1.08 1.08	004 004 004 004 004 004 004 004 004	362 362 362 362 362 362 362 362 362	700 700 700 700 700 700 700 700 700 700	014 018 014 014 014 014 014 014	236 236 236 236 236 236 236 236	418 672 668 648 468 458 458 458 458 458	3 00 7 00 3 00 3 00 7 00 7 00 7 00 7 00
Filter 1 Filter 2 Filter 3 Filter 4 Giar Web Divide Receival	Fibration System	000000	000000	51 52 55 55 56 57 58	MLD MLD MLD ML ML ML	0.29 0.29 0.29 0.29 0.29 0.29 0.29	051 051 051 051 051 007	0.62 0.62 0.62 0.62 0.69	0.80 0.60 0.60 0.60 0.60 0.79	-0.80 -0.80 -0.80 -0.80 -0.78 -0.79 -0.79	0.06 0.05 0.05 -6.07 -6.07	071 071 071 071 001 061	060 060 060 018 078	07 07 07 07 04 04 04	034 034 034 034 034 038 038	071 071 071 071 061	0.04 0.04 0.04 0.04 0.04 0.04	602 562 602 362 362	90.80 90.90 10.80 90.90 7.00 7.00	0.54 0.54 0.54 0.54 0.54 0.54	130 036 036 036 036 036	6.68 5.68 5.68 4.68 4.68	10:00 10:00 10:00 7:00 7:00
Persp 1 Persp 1 Persp 4 Persp 2	High Lifting System	00000	00000	2000	MLD MLD MLD MLD	017 017 017 017 017	0.29 0.29 0.29 0.29	0.30 0.30 0.30 0.30	0.94 0.94 0.94 0.94 0.94	018 018 019 019	1000 000 000 000 000 000 000 000 000 00	000 000 000 000	019 019 019 019	011 036 036 036	0.50 0.50 0.50 0.50 0.50	089 089 089 089 080	7.00 7.00 7.00 2.00	16.2 18.50 18.60 18.60 19.60	30 90 30 90 30 90 30 90	7 10 7 10 7 10 7 30	148 148 148 148 148	110 22.55 22.55 22.55	3000 3000 3000 3000
	Comeyence System	0 8	0 4		MLD	0.19	0.94	Q(34	0.40	Gπ	0.16	09(φn.	0.13	G39	0er i	0.08	0.42	6.75	a.18	0.31	051	0.75
Tecnini/Reservir 1	By when	D 9	a 1	65	MLD.	0.00	0.72 0.73	0.72 0.72	0.96 0.00	42.96 42.95	0.02 6.02	1.00	0% 0%	40 40	90.09 80 D	1.04 1.08	0.04 0.04	3.62 3.62	7.00 r.02	0.84 11.84	2.36 2.51	4.60 4.62	7.00 7.00

APPENDIX II: MATLAB SOURCE CODE II-A LAKE HURON PRIMARY WATER SUPPLY SYSTEM (LHPWSS)

```
C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmLH.m
                                                                                  Page 1
December 2, 2004
                                                                             4:41:32 PM
    % Last Modified: Thursday, October 07, 2004 @5:45pm
 1
 2
     x=-10:0.05:5;
 3
    time=0:0.1:200;
 4
    alpha=0:0.05:1;
 5 %%%......Step ONB.....Building Structure Array
 6 $$$it is an array that containes fields under each fieldthere are values or
 7
    ***strinngs
 A.
 9
         #determines the number of total elements in the system
10
         [No_of_Elements,Dummy01]=size(rowheaders);
11
1.2
         * Build the Structure array from the imported excel file.
13
         for n=1:No of Elements
14
                 System(n) = struct('Element_Name', rowheaders(n,1),'Element_ID',data(n,1 e)
     :4) . . . .
15
                     'Element_global_No', data(n, 5), 'Element_Capacity', data(n, 6:8), ...
16
                     'Element Requirment', data(n,9:11), 'Element Norm Cap Tri', data(n,12:1 #
     4), 'Element_Norm_Req_Tri', data(n, 15:17),...
17
                     'Element Norm Cap Trap', data (n, 18:21), 'Element Norm Reg Trap', data (n 🖌
     ,22:25),...
18
                     'Element State TriMos', data (n. 26:28), 'Element State TrapMos', data (n. 🖌
     29:32),...
19
                     'Element Failure Tri',data(n,33:35),'Element Failure Trap',data(n,38 🖌
     :39))/
20
         and
21
22
     #### ......Step TWO.....Deterimine the No. of Sub-Systems
23
     $8% by caclualting the maximum No. Value in the ID first two digits 44 Deterimine
24
     *** the No. of elements in each Sub-System by caclualting the maximum No. Value in
25
     *** the ID last two digits
26
27
             for sse-1:No of Elements
                                                 % sse stands for Sub-System Element
28
                 % Calculation of the Number of Sub-Systems
29
                 SubSystem (sse, 1) = [ [System (sse).Element ID(1, 1) System (sse).Element ID(1, 4
     2)]];
30
                 SubSystm Tenth(sss,1)=num2str(SubSystem{sss,1}(1,1));
31
                 SubSystm Units(sse, 1)=num2str(SubSystem{sse, 1](1,2));
32
                 SubSystem_String_Code=strcat(SubSystm_Tenth,SubSystm_Units);
33
                 SubSystem Numerical Code=str2num(SubSystem String Code);
34
                 Total SubSystems-max(SubSystem Numerical Code);
35
36
                 * Calculation of the Number of Elements in each Sub-System
```

C:\ Dec	Documents and Settings\Ibrahim\Desktop\Daily w\rmLH. ember 2, 2004	m Page 2 4:41:32 PM
37	Element(sse, 1)={(System(sse).Element_ID(1,3) System(sse	.Element_ID(1,4) 🖌
	1}/	
38	SubSystm_Element_Tenth(sse, 1) = num2str(Element{sse, 1}(1,	1)))
39	<pre>SubSystm_Element_Units(sse,1)=num2str(Element[sse,1](1,)</pre>	2));
40	Element_String_Code=strcat(SubSystm_Element_Tenth,SubSy	stm_Element_Units 🖌
	11	
41	Element_Numerical_Code=str2num(Element_String_Code);	
42	end	
43		
44	System_Numeric_ID_Array=[SubSystem_Numerical_Code Element_Numerical	_Code];
45		
46	* Creating zero matrix to store information on number of elements is	n each subsystem
47	Information_Array=zeros(Total_SubSystems,2);	
48		
49	counter=1;	
50	for sse2=1:(No_of_Elements-1)	
51	Information_Array(counter,1)=counter;	
52	Information_Array(counter,2)=System_Numeric_ID_Array(sse2,2)) x
53		
54	<pre>if (System_Numeric_ID_Array((sse2+1),2))<(System_Numeric_ID_Array())</pre>	_ID_Arrsy(sse2,2 🖌
	(1) (1) (2) (2) (2) (2) (2) (2) (2) (2) (2) (2	
55	counter=counter+1;	
56	and	
57		
58	if counterTotal_SubSystems	
59	Information_Array(counter,2)=System_Numeric_ID_Array	(sse2+1,2);
60	and	
61	end	
62		
63	***************************************	
64	\$\$\$Step THREEMake Puzzy MP usable in Augmentation \$	
65	<pre>\$2% Osing the Triangula MP in the system Matrix \$</pre>	
66	for i=1:No_of_Elements *	
67	A=System(1).Element_State_TriNos(1,1); 8	
68	B=System(1)_Element_State_TriMos(1,2); 8	
69	C=System(1) Element_State_TriNos(1,3); *	
20	Element_MF(1,1)={[triangular(alpha,A,B,C)]}; 8	
71	ena	
12	***************************************	
13		
74	***	
15	*** Using the Trapisoidal NF in the system Matrix *	

```
C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmLH.m
                                                                          Page 3
 December 2, 2004
                                                                     4:41:32 PM
 76
      %for i=1:No_of_Elements
                                                                ģ.
 77
             A=System(i) Blement State TrapMos(1,1);
                                                                8
       8
 78
              B-System(1) Blement_State_TrapMos(1,2);
                                                                ž
        5
 79
        1
             C=System(i).Element State TrapMos(1,3);
 80
         * D=System(i).Blement_State_TrapMos(1,4);
 81
          # Blement MP(1,1)=[[trapizoidal(alpha,A,B,C,D)]];
 32
          kond.
 83
     84
 86 ### using the SUMMATION operator for both
 87 $% NOTE: all redundant elements or parallel elelemtns will be added to the
 88 ** first element in the group (all values will be incorporated in the MF
 89 ** value of the first element)
 90
     statessestatessestatesses Redundant Elementssestatessestatessestatesse
 91
 92
                        %1. Intake Sub-System
 93
                              there is no redundancy in any of the elements
 94
 95
                        $2. Low Lifting Sub-System
 96
                              § ] Redundant elements
 97
                                   04----07. 05----08 & 06----09
 98
     Element MF{7,1}=fuzzymath(alpha,Element MF{7,1},Element MF{10,1},'sum');
 99
     Element MF{8,1}=fuzzymath(alpha,Element MF{8,1},Element MF{11,1},'sum');
100
     Element MF(9,1)=fuzzymath(alpha,Element MF(9,1),Element MF(12,1),'sum');
101
102
                        $3. Palch Mixing Sub-System
103
                            *PAC, Alum, and Polymer Sub system
104
                              % 1 Redundant element
105
                                  03----21
                               8
106
     Element_MF{15,1}=fuzzymath(alpha,Element_MF{15,1},Element_NF{33,1},'sum');
107
108
                        $4. Flocculation Sub-System
                               * there is no redundancy in any of the elements
109
110
111
                        $5. Sedimentation Sub-System
112
                               there is no redundancy in any of the elements
                                                                                1
113
114
                        ₹6. Piltering Sub-System
115
                               * there is no redundancy in any of the elements
                                                                                .
```

16	
17	 High Lifting Sub-System
18	% 2 Redundant elements
19	\$ 0104 & 0205
2.0	Element_MF{63,1}=fuzzymath(alpha,Element_MF{63,1},Element_NF{66,1},'sum');
21	Element_MP{64,1}=fuzzymath(alpha,Element_MP{64,1},Element_MP{67,1},'sum');
22	
23	 Conveyance Sub-System
2.4	* there is no redundancy in any of the elements
25	
26	 Boosting Sub-System
27	% there is no redundancy in any of the elements
28	
29	#10. Storage Sub-System
30	% there is no redundancy in any of the elements
31	
3,2	***************************************
	2222 C
33	**************************** Parallel Elements************************************
	8888
34	%1. Intake Sub-System
35	# there is no parallel elements
36	
37	\$2. Low Lifting Sub-System
38	% J Faralel elements
39	\$ 04-05-06
4.0	Element_MF{7,1}=fuzzymath(alpha,Element_MF{7,1},Element_MF{8,1},'sum');
41	<pre>Element_MF{7,1}=fuzzymath(alpha,Element_MF{7,1},Element_MF{9,1},'sum');</pre>
42	
43	 Palsh Mixing Sub-System
4.4	<pre>%DAC, Alum, and Dolymer Sub_system</pre>
45	% there is no parallel elements
46	Parallelsmis between the whole two lines not
47	% elements
48	
49	 Plocoulation Sub-System
50	% 2 Paralel groups of elements
51	¥ (01,03,05,07) \$ (02,04,06,08)

```
C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmLH.m
                                                                                   Page 5
 December 2, 2004
                                                                              4:41:32 PM
152
      Element MF{34,1}=fuzzymath(alpha,Element MF{34,1},Element MF{36,1},'sum');
153
      Element MP{34,1}=fuzzymath(alpha,Element NP{34,1},Element NP{38,1},'sum');
154
      Element_MP{34,1}-fuzzymath(alpha,Element_MP{34,1},Element_MP{40,1},'sum');
155
      Element MP{35,1}=fuzzymath(alpha,Element MP{35,1},Element MP{37,1},'sum');
156
      Element_MF{35,1}=fuzzymath(alpha,Element_MF{35,1},Element_MF{39,1},'sum');
157
      Element MP{35,1}=fuzzymath(alpha,Element NP{35,1},Element MP{41,1},'sum');
158
159
160
                           %5. Sedimentation Sub-System
161
                               % 2 Paralel groups of elements
162
                               $ (01,03) - (02,04)
163
      Element MF{42,1}=fuzzymath(alpha,Element MF{42,1},Element MF{44,1},'sum');
164
165
      Element_MF{43,1}=fuzzymath(alpha,Element_NF{43,1},Element_NF{45,1},'sum');
166
167
                           #6. Filtering Sub-System
158
                                  % 3 Paralel groups of elements
159
                                  $ (01,03,05,07,09,11) & (02,04,06,08,10,12) &
170
                                  $ (15, 16, 17)
171
      Element MF{46,1}=fuzzymath(alpha,Element NF{46,1},Element NF{48,1},*sum');
172
      Element MF{46,1}-fuzzymath(alpha,Element MF{46,1},Element MF{50,1},*sum');
173
      Element MF{46,1}-fuzzymath(alpha,Element MF{46,1},Element MF{52,1},*sum');
174
      Element MF{46,1}=fuzzymath(alpha,Element MF{46,1},Element MF{54,1},'sum');
175
      Element MF{46,1}=fuzzymath(alpha,Element NF{46,1},Element MF{56,1},'sum')/
176
      Element MF{47,1}=fuzzymath(alpha,Element MF{47,1},Element MF{49,1},'sum')/
177
      Element MF{47,1}-fuzzymath(alpha,Element NF{47,1},Element NF{51,1},'sum');
178
      Element MF{47,1}-fuzzymath(alpha,Element NF{47,1},Element MF{53,1},'sum');
179
      Element MP{47,1}-fuzzymath(alpha,Element NP{47,1},Element MP{55,1},'sum');
180
      Element MF{47,1}-fuzzymath(alpha,Element MF{47,1},Element MF{57,1},'sum');
181
      Element MP{60,1}=fuzzymath(alpha,Element MP{60,1},Element MP{61,1},'sum');
182
      Element_MF{60,1}=fuzzymath(alpha,Element_MF{60,1},Element_MF{62,1},'sum');
183
184
                           $7. High Lifting Sub-System
185
                                  $ 1 Paralel group
186
                                  8 (01,02,03)
187
      Element MF{63,1]=fuzzymath(alpha,Element MF{63,1},Element MF{64,1},'sum');
188
      Element MP{63,1}=fuzzymath(alpha,Element MP{63,1},Element MP{65,1},'sum');
189
190
                           $8. Conveyance Sub-System
191
                                  a I Paralel group
192
                                  $ (01,02)
193
      Element_MP{68,1}=fuzzymath(alpha,Element_MP{68,1},Element_MP{69,1},'sum');
                                                                                          16
```

C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmLH.m Page 6 December 2, 2004 4:41:32 PM

194 195 \$9. Boosting Sub-System 196 % 1 Paralel group. 197 \$ (01,02,03,04) 198 Element MP{70,1}=fuzzymath(alpha,Element NP{70,1},Element NP{71,1},'sum'); 199 Element_MP{70,1}=fuzzymath(alpha,Element_MP{70,1},Element_MP{72,1},'sum'); 200 Element_MF{70,1}=fuzzymath(alpha,Element_MF{70,1},Element_NF{73,1},'sum'); 201 202 \$10. Storage Sub-System 203 * 1 Paralel group 204 8 (01,02,03,04) 205 Element MF{75,1}=fuzzymath(alpha,Element MF{75,1},Element MF{76,1},'sum'); 206 Element_MF{75,1}=fuzzymath(alpha,Element_MF{75,1},Element_MF{77,1}, 'sum'); 207 Element MF{75,1}=fuszymath(alpha,Element MF{75,1},Element MF{78,1},*sun*); 208 22.23 209 **** 210 211 * Build the sequence of Augmentation Sub-System step wise 212 * Connection between sub_system is considered serial connection 213 214 %1. Intake Sub-System 215 # 1 serial 2 serial 3 216 System State-fuzzylogmath(x,alpha,Element MF{1,1},Element MF{2,1}, 'minimum'); 217 Systam State-fuzzylogmath(x, alpha, Systam State, Element MF[3,1], 'minimum'); 218 219 \$2. Low Lifting Sub-System 220 * 1 serial 2 serial 3 serial (4* parallel 5* parrallel 6*) 221 % star element means it has a rdundant element. 222 System State=fuzzylogmath(x,alpha,System State,Element MF(4,1),'minimum'); 223 System State=fuzzylogmath(x,alpha,System State,Element MF{5,1},'minimum'); 224 System State-fuzzylogmath(x,alpha,System State,Element MF(6,1),'minimum'); System State-fuzzylogmath(x,alpha,System_State,Element_MP{7,1},'minimum'); 225 226 227 83. Falsh Mixing Sub-System 228 %PAC, Alum, and Polymer Sub_system 229 § (1 serial 3* serial 5 serial 7 serial 9 serial 11 230 serial 13 serial 15 serial 17 serial 19) 8 231 & PARALLEL TO 23.2 % (2 serial 4 serial 6 serial 8 serial 10 serial

-	emper 2, 2003 3131.35 1
233	\$ 12 serial 14 serial 16 serial 18 serial 20)
234	% this summation to avoid producing MPs that can not be summed using the
235	% summation function (as minmising will produce MPs in different froms
236	% than that needed by the summation function)
237	<pre>Element_MF{13,1}=fuzzymath(alpha,Element_MF{13,1},Element_MF{14,1},'sum');</pre>
238	Element MF{15,1}=fuzzymath(alpha,Element MF{15,1},Element MF{16,1},'sum');
23.9	Element MF{17,1}=fuzzymath(alpha,Element_MF{17,1},Element_MF{18,1},'sum');
240	<pre>Element_MF{19,1}=fuzzymath(alpha,Element_MF{19,1},Element_MF{20,1},'sun');</pre>
241	Element MF{21,1}=fuzzymath(alpha,Element MF{21,1},Element MF{22,1}, "sun");
242	Element_MF{23,1}=fuzzymath(alpha,Element_MF{23,1],Element_MF{24,1},'sun');
243	Element_MF{25,1}=fuzzymath(alpha,Element_MF{25,1},Element_MF{26,1},'sum');
24.4	Element MF{27,1}=fuzzymsth(alpha,Element MF{27,1},Element MF{28,1},'sum');
245	<pre>Element_MF{29,1}=fuzzymath(alpha,Element_MF{29,1},Element_MF{30,1},'sum');</pre>
246	Element_MF{31,1}=fuzzymath(alpha,Element_MF{31,1},Element_MF{32,1},'sum');
247	Element MF{13,1}=fuszylogmath(x,alpha,Element MF{13,1],Element MF{15,1},'minimum')
248	Element_MF{13,1}=fuzzylogmath(x,alpha,Element_MF{13,1},Element_MF{17,1},'minimum')
249	Element MF{13,1}=fuzzylogmath(x,alpha,Element MF{13,1},Element MF{19,1},'minimum')
250	Element MF{13,1}=fuzzylogmath(x,alpha,Element MF{13,1},Element MF{21,1},'minimum')
251	Element_MF{13,1}=fussylogmath(x,alpha,Element_MF{13,1},Element_MF{23,1},'minimum')
252	<pre>Element_MF{13,1}=fuzzylogmath(x,alpha,Element_MF{13,1},Element_MF{25,1},'minimum')</pre>
253	Element_MF{13,1}=fuzzylogmath(x,alpha,Element_MF{13,1},Element_MF{27,1},'minimum')
254	<pre>Element_MF{13,1}=fuzzylogmath(x,alpha,Element_MF{13,1},Element_MF{29,1},'minimum')</pre>
255	<pre>Element_MF{13,1}=fuzzylogmath(x,alpha,Element_MF{13,1},Element_MF{31,1},'minimum')</pre>
256	
257	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{13,1},'minimum');
258	
259	#4. Plocculation Sub-System
260	% no serials, only previous system-state and
261	% sum of tow parrallel groups of elements
262	<pre>Element_MP{34,1}=fuzzymath(alpha,Element_MP{34,1},Element_MP{35,1},'sum');</pre>
263	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{34,1},'minimum');
264	
265	 Sedimentation Sub-System
266	\$ no serials, only previous system-state and
267	% sum of tow parrallel groups of elements
268	<pre>Element_MF{42,1}=fuzzymath(alpha,Blement_MF{42,1},Blement_MF{43,1},'sum');</pre>
269	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{42,1},'minimum');
270	
271	%6. Filtering Sub-System
272	<pre>% (01,03,05,07,09,22) serial 13</pre>
273	* PARALLEL TO

-	
275	* SERIAL TO (15*, 16, 17)
276	<pre>Element_MF{46,1}=fuzzymath(alpha,Element_MF{46,1},Element_NF{47,1},'sum');</pre>
277	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{46,1},'minimum');
278	<pre>Element_MF{58,1}=fuzzymath(alpha,Element_MF{59,1},Element_MF{59,1},'sum');</pre>
279	System_State=fuzzylogmath(x,alpha,System_State,Element_MF(58,1},'minimum');
280	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{60,1},'minimum');
281	
28.2	 Wigh Lifting Sub-System
283	* no serials, only previous system-state
284	System_State=fuzzylogmath(x,alpha,System_State,Element_MF[63,1],'minimum');
285	
286	*8. Conveyance Sub-System
285	* no serials, only previous system-scare
200	System_State=ruzzyrogmath(x,arpha,System_State,Erement_MP(08,1), 'minimum');
290	89 Roosting Sub-System
291	\$ (03 02 03 04) serie] 5
292	System State_fuszylogmath(x_alpha_System State Element ME(70.1) !minimum!);
293	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{74,1},'minimum');
294	
295	%10. Storage Sub-System
29.6	this sub-system is paralle to the previous
297	1 sub-system
298	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{75,1},'minimum');
299	
300	[Row_x,Col_x]=size(x); <pre>% basic information about x-values (x-axis)</pre>
100	NO_X_BCEDB=CDI_X-1;
302	X_step=max(x)/No_X_steps;
203	A fearing the priving? Sustem State ME with same for farther use in
205	a Acepting the original system state of with serve to farther use in
306	Guotam Stata Vulba_Svetam Stata;
307	22222222222222222222222222222222222222
20.8	System State-(System State/Find(System State(- 2)-50) 1) System State/Find(System State)
	te(:,2)>0),2)];
809	[dumm_row,dumm_col]=size(System_State);
310	System_State=[(System_State(1,1)-x_step) 0;System_State;(System_State(dumm_row,1)+x_)
	step) 01;
311	
312	\$\$\$\$\$\$\$\$\$\$\$Svstem State Areas\$

C:\ Dec	pocuments and settings/ibranim/pesktop/paily w/rmLH.m Page ember 2, 2004 4:41:32 P	9 M
212	2 Determine the maximum alpha value is teh sustem state matrix	-
214	Marimum alpha System State-mar(System State(- 2)).	
315	Harimon_arbua_chacan_crace-marichacan_crace()/c///	
216	& based on the No. of rows, a loop will search for the row inder of the	
317	<pre>% maximum almha value</pre>	
318	lyou Svetem State column Svetem Statel_eize(Svetem State).	
319	max index System State=0:	
32.0	for t=1:row System State	
321	max index System State-max index System State+1;	
322	if System State(t,2)==Maximum alpha System State	
323	break /	
324	end	
325	end	
326		
327	% the range from the maximum value to Zero will be divided for high	
328	* resolution alpha.	
329	alpha range System State-Maximum alpha System State;	
330	step no System State=100;	
331	step_System_State=Maximum_alpha_System_State/step_no_System_State;	
332		
333	<pre>%Interpolation</pre>	
334	%Left-Hand Interpolation	
335	Left_System_State=(interpl(System_State(1:max_index_System_State,2),System_State,2)	
	em_State(1:max_index_System_State,1),0:step_System_State:Maximum_alpha_System_State)	1¢
	•) /	
336	System_State_L=[Left_System_State (0:step_System_State:Maximum_alpha_Sys	3 ¥
	tem_State)'];	
337	<pre>%Right-Hand Interpolation</pre>	
338	Right_System_State=(interpl(System_State(max_index_System_State:row_System_State))	: *
	em_State,2),System_State(max_index_System_State:row_System_State,1),0:step_System_St	. 4
	ate:Maximum_alpha_System_State) *);	
339	System_State_R~[Right_System_State (0:step_System_State:Maximum_alpha_S)	14
	<pre>stem_State) '];</pre>	
340		
341	%Area-Calculation	
342	strip_Area_System_State=0;	
343	<pre>for tt=1:(step_no_System_State-1)</pre>	
344	strip_Area_System_State=strip_Area_System_State+(0.5*((System_State_R(tt,1)	K
	System_State_L(tt,1))+(System_State_R((tt+1),1)-System_State_L((tt+1),1)))*step_System_State_L((tt+1),1)))*step_System_State_L(tt+1),1))	. 4
	em_State);	
345		
346	Area_System_State=strip_Area_System_State+(0.5*(System_State_R((step_no_System_)	3 K

```
C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmLH.m
                                                               Page 10
 December 2, 2004
                                                            4:41:32 PM
     tate+1),1)-System_State_L((step_no_System_State+1),1))*step_System_State);
347
348
        Weighted Area-Calculation
349
        weighted strip Area System State=0;
350
        for ttt=1: (step_no_System_State-1)
351
           weighted strip Area System State-weighted strip Area System State+(10.5*((Sy 4
     sten_State_R(ttt,1)-System_State_L(ttt,1))+(System_State_R((ttt+1),1)-System_State_L 🖌
     ((ttt+1),1)))*step_System_State)*((System_State_L(ttt,2)+System_State_L((ttt+1),2))* #
     0.5))/
352
        end
353
        Weighted_Area_System_State=weighted_strip_Area_System_State+((0.5*(System_State_ *
     R((step no System State+1),1)-System State L((step no System State+1),1))*step Syste 4
     m State)*System State L(step no System State,2));
354
        355
356
    357
    *** ......Step PIVE.....Define the Acceptable level(s) of performance
358 *** using trapizoidal MF as (a,b,c,d) where c=d
                                                               3
359
    No of Levels=3;
                                                               3
360 m1=0.6; m2=0.7; m3=5; m4=5;
                                                               3
361 Level 1y=trapmf(x,[m1 m2 m3 m4]);
362 Level 1=(x' Level 1y');
363 LR Level I=(m1*m2)/(m2-m1);
364
365 nl=0.6; n2=1.2; n3=5; n4=5;
                                                               1
366 Level 2y=trapmf(x, [n1 n2 n3 n4]);
367 Level 2=[x' Level 2y'];
368 LR Level 2= (n1*n2) / (n2-n1) /
369
370 q1=0.6; q2=5; q3=5; q4=5;
                                                              8
371 Level 3y=trapmf(x,[q1 q2 q3 q4]);
372 Level 3=[x' Level 3y'];
373 LR Level 3=(q1*q2)/(q2-q1);
374
375 LR=[LR Level 1;LR Level 2;LR Level 3];
376
380 $8% of the state function and each level
381 OverLap Level 1y=min(Level 1(:,2),System State xwise(:,2));
382 OverLap_Level_1=[x' OverLap_Level_1y];
```

C:\ Dec	pocuments and settings/ibranim/besktop/baily w/rmLH.m Page 11 ember 2. 2004 4:41:32 PM
2.82	
384	OverLap Level 2v=min(Level 2(; 2),Svetem State xwise(; 2));
385	OverLap Level 2-[x' OverLap Level 27]
386	
387	OverLap Level 3y=min(Level 3(:,2),System State xwise(:,2));
388	OverLap Level 3=[x' OverLap Level 3y]
389	
390	* take-off the zeros from both ends of the overlap area
391	
392	<pre>Area_Level_1=OverLap_Level_1((find(OverLap_Level_1(:,2)>0)),:);</pre>
3.93	Area_Level_1_start=(min(OverLap_Level_1((find(OverLap_Level_1(:,2)>0)),1))-x_step 0] #
394	<pre>Area_Level_1_end=[max(OverLap_Level_1((find(OverLap_Level_1(:,2)>0)),1))+x_step 0];</pre>
395	Area_Level_1=[Area_Level_1_start;Area_Level_1;Area_Level_1_end];
396	
397	<pre>Area_Level_2=OverLap_Level_2((find(OverLap_Level_2(;;2)>0));;);</pre>
398	<pre>Area_Level_2_start=[min(GverLap_Level_2((find(OverLap_Level_2(:,2)>0)),1))-x_step 0] #</pre>
399	, Area Level 2 end=[max(OverLap Leve] 2((find(OverLap Level 2(, 2)>0)),1))+x step 0],
400	Area Level 2- [Area Level 2 start Area Level 2 Area Level 2 end];
401	
402	
403	Area Level 3-OverLap Level 3((find(OverLap Level 3(:,2)>0)),:);
404	Area_Level_3_start=[min(OverLap_Level_3((find(OverLap_Level_3(:,2)>0)),1))-x_step 0] #
405	<pre>Area_Level_3_end=(max(OverLap_Level_3((find(OverLap_Level_3(:,2)>0)),1))+x_step 0];</pre>
406	Area_Level_3= [Area_Level_3_start; Area_Level_3; Area_Level_3_and];
407	
408	***************************************
409	<pre>%%%Step SEVENCalculation of the overlap area & weighted area by inte # prolation over alpha cuts</pre>
410	*** for each overlap matrix
411	
412	\$\$\$\$\$\$\$\$\$Level ONE\$
413	* Determine the maximum alpha value in teh ovelap matrix.
414	<pre>Maximum_alpha_Area_Level_1=max(Area_Level_1(:,2));</pre>
415	
416	% based on the No. of rows, a loop will search for the row index of the
417	* maximum alpha value
418	[row_Area_Level_1,column_Area_Level_1]=size(Area_Level_1);
419	max_index_Area_Level_1=0;
420	for ii=1:row_Area_Level_1

Dec	ember 2, 2004 4:41:32 PM
421	max index Area Level 1=max index Area Level 1+1;
422	if Area_Level_1(ii,2)==Maximum_alpha_Area_Level_1
423	break,
424	end
425	end
426	
427	* the range from the maximum value to Zero will be divided for high
428	* resolution alpha.
429	alpha range Area Level 1-Maximum alpha Area Level 1;
430	step no Area Level_1=100;
431	step_Area_Level_1-Maximum_alpha_Area_Level_1/step_no_Area_Level_1/
432	
433	*Interpolation
434	%Left-Hand Interpolation
435	Left Area Level 1=(interpl(Area Level 1(1:max index Area Level 1,2),Area •
	_Level_1(1:max_index_Area_Level_1,1),0:step_Area_Level_1:Maximum_alpha_Area_Level_1) •
	*) c
436	Area Level 1 L=[Left Area Level 1 (0:step Area Level 1:Maximum alpha Are •
	a Level 1)');
437	%Right-Hand Interpolation
438	Right Area Level 1=(interpl(Area Level 1(max index Area Level 1:row Area •
	Level 1,2),Area Level 1(max index Area Level 1:row Area Level 1,1),O;step Area Leve .
	l 1.Maximum alpha Area Level 1)*);
43.9	Area Level 1 R-[Right Area Level 1 (0:step Area Level 1:Maximum alpha Ar •
	ea Level 1)' /
440	
441	*Area-Calculation
44.2	strip Area Area Level 1-0;
443	for jj=1: (step no Area Level 1-1)
444	strip Area Area Level 1=strip Area Area Level 1+(0.5*((Area Level 1 R(jj,1)- •
	Area Level 1 L(j),1)+(Area Level 1 R((jj+1),1)-Area Level 1 L((jj+1),1))*step Area .
	Level 1);
445	end
446	Area Area Level 1-strip Area Area Level 1+(0.5*(Area Level 1 R((step no Area Lev •
	el 1+1),1)-Area Level 1 L((step no Area Level 1+1),1))*step Area Level 1);
447	
44.8	Weighted Area-Calculation
449	weighted strip Area Area Level 1=0;
450	for kk=1: (step no Area Level 1-1)
451	weighted strip Area Area Level 1=weighted strip Area Area Level 1+((0.5*((Ar *
	ea Level 1 R(kk, 1) - Area Level 1 L(kk, 1)) + (Area Level 1 R((kk+1), 1) - Area Level 1 L((k +
	bill 100 weter area level 1040 (Area Level 1 1/66 2) area Level 1 1/66441 20040 50).

Dec	ember 2, 2004 4:41:32 PM
452	and
452	Weichted åres äres Level 1-weichted strin åres äres Level 12000 500 åres Level 1
100	R((step_no_Area_Level_1+1),1)-Area_Level_1_L((step_no_Area_Level_1+1),1))*step_Area_ &
404	Level_1)*Area_bevel_1_b(step_no_Area_bevel_1,2));
454	ANALISASI,, DOAD OWP'''' ALLERANGERENEERSEERSEERSEERSEERSEERSEERSEERSEERS
455	
420	essessesses,,Level ING.,.essessessessessessessessesses
427	* Decenmine che maximum alpha varue in can overap matrix.
458	Maximum_alpha_Area_Level_2=max(Area_Level_2(1,2));
45.9	men formand i secondo se eta mande incorrente en el tradat des 814 desenvoltes deste material des formandes des
450	* nased on the No. of rows, a loop will search for the row index of the
451	* maximum alpha value
402	[IOW_Area_Level_2,Column_Area_Level_2]=Size(Area_Level_2);
403	max_fidex_area_Level_2=0;
464	for 111=1:row_Area_Level_2
400	max_index_Area_Level_z=max_index_Area_Level_z+1;
455	if Area_Level_2(111,2)==Paximum_alpha_Area_Level_2
467	break;
458	end
469	end
470	
471	% the range from the maximum value to Zero Will be divided for high a second to be a second t
414	* resolution alpha.
473	aipna_range_Area_Levei_z=Maximum_alpna_Area_Levei_z;
474	step_no_Area_Level_2=100;
475	step_Area_Level_2=Maximum_aipna_Area_Level_2/step_ho_Area_Level_2/
476	
411	*Interpolation
478	#Left-Hand Interpolation
479	Left_Area_Level_2=(interpl(Area_Level_2(1:nax_index_Area_Level_2,2),Area ¥
	_Level_2(1:max_index_Area_Level_2,1),0:scep_Area_Level_2:Maximum_alpha_Area_Level_2) &
480	Area_Level_2_L=(Left_Area_Level_2_(0:step_Area_Level_2:Maximum_alpha_Are #
	a_Level_2)'l;
481	skight-Hand Interpolation
482	Right_Area_Level_2=(interp1(Area_Level_2(max_index_Area_Level_2:row_Area #
	Level 2,2), Area Level 2(max_index_Area_Level 2:row_Area_Level 2,1), 0:step_Area_Leve &
122	1_2:Maximum_alpha_Area_Level_2)');
483	Area_Level_2_R=(Right_Area_Level_2 (0:step_Area_Level_2:Maximum_alpha_Ar ¥
4.0.4	ea_Level_2/14
484	
485	MArea-Calculation

Dec	ember 2, 2004 4:41:32 PM
186	strip Area Area Level 2=0:
187	for jj=1:(step no Area Level 2-1)
188	strip Area Area Level 2-strip Area Area Level 2+(0.5*/(Area Level 2 R()), 1)
	-Area Level 2 L(iii))+(Area Level 2 R((iii+1),1)-Area Level 2 L((iii+1),1)))*step
	Area Level 2);
189	end
190	Area Area Level 2=strip Area Area Level 2+(0.5*(Area Level 2 R((step no Area Lev
	el 2+1),1)-Area Level 2 L((step no Area Level 2+1),1))*step Area Level 2);
191	
192	Weighted Area-Calculation
193	weighted strip Area_Area_Level_2~0,
194	for kkk=1:(step no Area Level 2-1)
195	weighted_strip_Area_Area_Level_2=weighted_strip_Area_Area_Level_2+((0.5*((Ar
	ea_Level_2_R(kkk,1)-Area_Level_2_L(kkk,1))+(Area_Level_2_R((kkk+1),1)-Area_Level_2_L
	((kkk+1),1)))*step_Ares_Level_2)*((Ares_Level_2_L(kkk,2)+Ares_Level_2_L((kkk+1),2))*
	0.5));
96	end
197	Weighted_Area_Area_Level_2=weighted_strip_Area_Area_Level_2+((0.5*(Area_Level_2_
	R((step_no_Area_Level_2+1),1)-Area_Level_2_L((step_no_Area_Level_2+1),1))*step_Area_
	Level_2) *Area_Level_2_L(step_no_Area_Level_2,2));
198	899888888888Level TWO888888888888888888888888888888888
199	
500	###############Level THREE################################
501	* Determine the maximum alpha value in teh ovelap matrix.
502	Maximum_alpha_Area_Level_3=max(Area_Level_3(:,2));
503	
504	* based on the No. of rows, a loop will search for the row index of the
505	* maximum alpha value
506	[row_Area_Level_3,column_Area_Level_3]=size(Area_Level_3);
507	max_index_Area_Level_3=0;
508	for iiii-1:row_Area_Level_3
609	<pre>max_index_Area_Level_3=max_index_Area_Level_3+1;</pre>
510	<pre>if Area_Level_3(iiii,2)==Maxinum_alpha_Area_Level_3</pre>
511	break /
512	and
513	end
514	
515	% the range from the maximum value to Zerc will be divided for high
516	<pre>% resolution alpha.</pre>
517	alpha_range_Area_Level_3=Maximum_alpha_Area_Level_3;
518	step_no_Area_Level_3=100;
519	step Area Level 3=Maximum alpha Area Level 3/step no Area Level 3;

Dec	ember 2. 2004 4:41:32 PM
520	
521	%Toterpolation
522	Maft. Wand Internalation
522	Left Bres Level 3- (interni (àres Level 3/1-new index àres Level 3 2) àres &
565	_Level_3(1:max_index_Area_Level_3,1),0:step_Area_Level_3:Maximum_alpha_Area_Level_3) &
	*);
524	Area_Level_3_L={Left_Area_Level_3 (0:step_Area_Level_3:Maxinum_alpha_Are 4
	a_Level_3)'];
525	<pre>%Right-Hand Interpolation</pre>
526	Right_Area_Level_3=(interp1(Area_Level_3(max_index_Area_Level_3:row_Area 4
	_Level_3,2),Area_Level_3(max_index_Area_Level_2;row_Area_Level_3,1),0;step_Area_Leve #
	1_3:Maximum_alpha_Area_Level_3)');
527	Area_Level_3_R=[Right_Area_Level_3 (0:step_Area_Level_3:Maximum_alpha_Ar 🗸
	ea_Level_3)'];
528	
529	*Area-Calculation
530	strip_Area_Area_Level_3=0;
531	for jjjj-1:(step_no_Area_Level_3-1)
532	strip_Area_Area_Level_J=strip_Area_Area_Level_J+(0.5*({Area_Level_J_R(jjjj,1 🖌
)-Area_Level_3_L(jjjj,1))+(Area_Level_3_R((jjjj+1),1)-Area_Level_3_L((jjjj+1),1)))*s 化
	tep_Area_Level_3) /
533	end
534	Area_Area_Level_3-strip_Area_Area_Level_3+(0.5*(Area_Level_3_R((step_no_Area_Lev 化
	el_3+1),1)-Area_Level_3_L((step_no_Area_Level_3+1),1))*step_Area_Level_3);
535	
536	Weighted Area-Calculation
537	weighted_strip_Area_Area_Level_3=0;
538	for kkkk=1:(step_no_Area_Level_3-1)
539	weighted_strip_Area_Area_Level_3=weighted_strip_Area_Area_Level_3+(10.5*(1Ar 🖌
	ea_Level_3_R(kkkk,1)-Area_Level_3_L(kkkk,1))+(Area_Level_3_R((kkkk+1),1)-Area_Level_ #
	3_L((kkkk+1),1)))*step_Area_Level_3)*((Area_Level_3_L(kkkk,2)+Area_Level_3_L((kkkk+1 #
);2))*0.5));
540	end
541	Weighted_Area_Area_Level_3=weighted_strip_Area_Area_Level_3+((0.5*(Area_Level_3_*
	R((step_no_Area_Level_3+1),1)-Area_Level_2_L((step_no_Area_Level_3+1),1))*step_Area_ &
	Level_3)*Area_Level_3_L(step_no_Area_Level_3,2));
542	\$\$\$\$\$\$\$\$\$\$Level THREE
543	
544	Area=[Area_Area_Level_1;Area_Area_Level_2;Area_Area_Level_3];
545	Max_Level=find(Area=-max(Area));
546	***************************************
F 1 12	

547 ***Step EIGHT.....Reliability measure Calculation

C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmLH.m Page 16 December 2, 2004 4:41:32 PM 548 CM Level 1=Weighted Area Area Level 1/Weighted Area System State; 549 CM Level 2=Weighted Area Area Level 2/Weighted Area System State; 550 CM_Level_2-Weighted_Area_Area_Level_3/Weighted_Area_System_State; 551 CM=[CM Level 1;CM Level 2;CM Level 3]; 552 553 Reliability Index=(max(CN)*LR(Max Level, 1))/max(LR) 554 556 Robustness Index 1 2=1/(CM Level 1-CM Level 2) 557 Robustness_Index_1_3=1/(CM_Level_1-CM_Level_3) 558 Robustness Index 2 3-1/(CM Level 2-CM Level 3) SG1 assessant states and state 562 *********************************** 563 *********************** 564 ************ 565 ****** 566 567 88 568 ₹ 569 38 570 \$88 ******* 571 572 ************ 577 ***Step TEN.....Reslience measure Calculation 579 ***Step TEN-ONE.....Make Puzzy MP usable in augmentation * 580 *** Using the Triangula MF in the system Matrix 2 for i=1:No of Elements 581 \$ 2 582 8 a-System(i).Element Failure Tri(1,1); 8 583 8 b=System(i).Element Failure Tri(1,2); 8 584 C=System(1).Element Pailure Tri(1,3)) 83 525 8 Blement_Pail(i,1)={{triangular(alpha,a,b,c)}}; 8 586 \$ and * 588 589 ***Step TEN-TWO..... Make Fuzzy MF usable in Augmentation *

0.0	Make Theirs the Consideridal MD is the sustan Materia	
120	for i-1-No of Elements	*
:02	ade lerind_of_blements	5
00	h-System(i) Element Failure Tran(1,2).	*
94	c-System(i) Element Failure Tran(1,3),	
95	A_Crotem(1) Element Failure Tran(1.4)	5
30	Elamant Pail(i 1) - [[tranizoida](alpha a b c d)]].	
97	end	
90	****	******
99	222 Cton TEM_TEPEE	undant alomonto
50	332 neize the MAYIMUM onerstor for both	WINGUP SIEWEILS
0.1	st more, all redundant elements or narallal alelemens will be a	dded to the
02	We first alamost in the group (all values will be incornerated	in the MP
50	## value of the first element)	A.S. 01302 003
04		
05	************************	****
06	\$7 Intake Sub-System	********
07	 There is no redundance in any of 	the elements
08	s more to no required in any or	
09	32 Low Lifting Sub-System	
10	% 7 Redundant elements	
11	8 0407 0508 6 0609	
12	Element Fail(7, 1)=fuzzylogmath(time,alpha Element Fail(7, 1),Ele	ment Fail(10.1).*maxi
10120	mim');	99779 4 09778978978788978897
13	Element_Fail(8,1}=fuzzylogmath(time,alpha,Element_Fail(8,1),Elemm");	ment_Fail{11,1},'maxi
14	Element Fail(9 1)=fuzzylogmath(time alpha Element Fail(9 1) Ele	mant Pail[12 1] *maxi
	uruu, / :	
15		
16	\$3. Palab Mixing Sub-System	
17	*PAC Alum, and Polymer Sub system	
18	* I Redundant element	
19	0321	
20	Element Fail 15,1 = fuzzylogmath (time, alpha, Element Fail (15,1), E	lement Fail{33,1},'ma
	ximum');	
21		
22	\$4. Plocoulation Sub-System	
23	* there is no redundancy in any of	the elements
24		
2.5	 Sedimentation Sub-System 	
200	a store in a sector of a sector of a sector of	the elements

0.00		
520	27	BILLING MA ALLE
248	\$0.	ritering sub-system
23.3		* there is no redundancy in any of the elements
(2.1	\$7	Vieb Liffing Cub Combon
131		high Litting Sub-System
54		\$ 2 Redundant elements
53		
34	<pre>ximum*);</pre>	ogmach(time,aipna,biement_raii[63,1],biement_raii[66,1], 'ma 🕷
35	<pre>Element_Pail(64,1}=fuzzylo ximun');</pre>	ogmath(time,alpha,Element_Fail{64,1},Element_Fail{67,1},'ma 🕊
36		
37	¥8.	Conveyance Sub-System
3.8		* there is no redundancy in any of the elements
39		
40	公司等	Boosting Sub-System
41		* there is no redundancy in any of the elements
43		
43	810.	Storage Sub-System
44		% there is no redundancy in any of the elements
45		
46	***********	***************************************
547	***************************************	Parallel Elements ####################################
4.9	81.	Intake Sub-System
49		there is no parallel elements
50		
51	\$2.	Low Lifting Sub-System
52		Paralel elements
53		8 04-05-06
54	Element_Fail{7,1}=fuzzylog	gmath(time,alpha,Element_Fail{7,1},Element_Fail{8,1},*maxim •
55	Element_Fail{7,1}=fuzzylog	gmath(time, alpha, Element_Fail{7,1}, Element_Fail{9,1}, *maxim •
2.2	mii 15	
57		
50		De Tale Windows Oak Standard
50	#35 1	Laren wirtud 200-slamer one instem
and the second second		srac, alum, and Polymer Bub_syStem
50		& these is as as 21-7 slopets

C:\ Dec	Documents and Settings\Ibrahim\Desktop\Daily w\rmLH.m Page 19 ember 2. 2004 4:41:32 PM
662	4 GIEDEUCA
600	the mission of the method
665	 Protociation sub-system 2 Paralal groups of elements
666	\$ (01.03.05.07) \$ (02.04.06.08)
667	<pre>Element_Fail{34,1}=fuszylogmath(time,alpha,Element_Fail{34,1},Element_Fail{36,1},'ma # ximum');</pre>
668	<pre>Element_Fail{34,1}=fuzzylogmath(time,alpha,Element_Fail{34,1},Element_Fail{38,1},'ma < ximum');</pre>
669	Element_Fail{34,1}=fuzzylogmath(time,alpha,Element_Fail{34,1},Element_Fail{40,1},'ma < ximum');
670	<pre>Element_Fail{35,1}=fuzzylogmath(time,alpha,Element_Fail{35,1},Element_Fail{37,1},'ma </pre> ximum*);
671	Element_Fail{35,1}=fussylogmath(time,alpha,Element_Fail{35,1},Element_Fail{39,1},'ma < ximum');
672	<pre>Element_Fail{35,1}=fuzzylogmath(time,alpha,Element_Fail{35,1},Element_Fail{41,1},'ma < ximum');</pre>
673	
674	\$5. Sedimentation Sub-System
675	% 2 Paralel groups of elements
676	8 (01,03) - (02,04)
677	Element_Fail{42,1}=fuzzylogmath(time,alpha,Element_Fail{42,1},Element_Fail{44,1},'ma & ximum');
678	<pre>Element_Fail{43,1}=fuzsylogmath(time,alphs,Element_Fail{43,1},Element_Fail{45,1},'ma < ximum');</pre>
679	
680	
681	 Piltering Sub-System
682	% 3 Paralel groups of elements
683	<pre>% (01,03,05,07,09,11) & (02,04,06,08,10,12) &</pre>
684	<pre>% /15,16,17)</pre>
685	<pre>Element_Fail{46,1}=fuszylogmath(time,alpha,Element_Fail{46,1},Element_Fail{48,1},'ma e ximum');</pre>
686	<pre>Element_Fail{46,1}=fussylogmath(time,alpha,Element_Fail{46,1},Element_Fail{50,1},'ma # ximum');</pre>
687	<pre>Element_Fail{46,1}=fuzzylogmath(time,alpha,Element_Fail{46,1},Element_Fail{52,1},'ma w ximum');</pre>
688	<pre>Element_Fail[46,1]=fuzzylogmath(time,alpha,Element_Fail{46,1},Element_Fail{54,1},'ma < ximum');</pre>
689	<pre>Element_Fail{46,1}=fussylogmath(time,alpha,Element_Fail{46,1},Element_Fail{56,1},'ma < ximum');</pre>
690	Element_Fail(49,1}=fuzzylogmath(time,alpha,Element_Fail(49,1},Element_Fail(51,1},'ma *

C:\ Dec	Documents and Settings\Ibrahim\Desktop\Daily w\rmLH.m Page 20 ember 2, 2004 4:41:32 PM
362	simm*):
691	Element_Fail{49,1}=fuzzylogmath(time,alpha,Element_Fail{49,1},Element_Fail{53,1},'ma w
692	Element_Fail{49,1}=fuzzylogmath(time,alpha,Element_Fail{49,1},Element_Fail{55,1},'ma w
693	Element_Fail(49,1)=fuzzylogmath(time,alpha,Element_Fail(49,1),Element_Fail(57,1),'ma w
694	Element_Fail(60,1}=fuzzylogmath(time,alpha,Element_Fail(60,1},Element_Fail(61,1},'ma w
695	Element_Fail[60,1}=fuzzylogmath(time,alpha,Element_Fail[60,1},Element_Fail[62,1},'ma w ximum');
696	
697	\$7. High Lifting Sub-System
698	% 1 Paralel group
699	¥ (01,02,03)
700	<pre>Element_Fail{63,1}=fuzzylogmath(time,alpha,Element_Fail{63,1},Element_Fail{64,1},'ma w ximum');</pre>
701	<pre>Element_Fail{63,1}=fuzzylogmath(time,alpha,Element_Fail{63,1},Element_Fail{65,1},'ma <</pre>
702	-FRAITCORD E E
703	%8. Conveyance Sub-System
704	\$ 1 Paralel group
705	\$ (01.02)
706	Element_Fail{68,1}=fuzsylogmath(time,alpha,Element_Fail{68,1},Element_Fail{69,1},'ma = ximum*);
707	
708	
709	 Boosting Sub-System
710	% 1 Paralel group
711	\$ (01,02,03,04)
712	<pre>Element_Fail{70,1}=fuzzylogmath(time,alpha,Element_Fail{70,1},Element_Fail{71,1},'ma # ximum');</pre>
713	<pre>Element_Fail{70,1}=fuzzylogmath(time,alpha,Element_Fail{70,1},Element_Fail{72,1},'ma * ximum*);</pre>
714	Element_Fail{70,1}=Euzzylogmath(time,alpha,Element_Fail{70,1},Element_Fail{73,1},'ma w ximum');
715	
716	%18. Storage Sub-System
717	<pre>% 1 Paralel group</pre>
718	¥ (01,02,03,04)
719	<pre>Element_Fail(75,1}=fuzzylogmath(time,alpha,Element_Fail(75,1],Element_Fail(76,1],'ma w ximum');</pre>

C:\ Dec	C:\Documents and Settings\Ibrahim\Desktop\Daily w\rmLH.m Page 21 December 2, 2004 4:41:32 PM		
720	Element_Fail{75,1}=fuzzylogmath(time,alpha,Element_Fail{75,1},Element_Fail{77,1},'ma < ximum');		
721	<pre>Element_Fail{75,1}=fussylogmath(time,alpha,Element_Fail{75,1},Element_Fail{73,1},'ma < ximum');</pre>		
722	*****		
723	**************************************		
724	[Row_time,Col_time]=size(time);		
726	% Rendering all Element-Pailure MPs a full length MP on time domain. for elabeled of Flemente.		
728	Element_Fail{e1,1}=fuzzylogmath(time,alpha,Element_Fail{e1,1},Element_Fail{e1,1} < .'maximum');		
729 730	end		
731. 732	* Thengetting the start and the end of the support of each Element-Failure MF * 3. Start of Support		
733	for s2=1:No_of_Elements		
735	<pre>if (Element_Fail(s2,1)(f,2)==0) & (Element_Fail(s2,1)((f+1),2)>0)</pre>		
737	end		
738 739	end		
740 741	# 2. End of Support for s3=1:No_of_Elements		
742 743	<pre>for f=1:(Col_time-1) if (Element_Fail{s3,1}(f,2)>0) & (Element_Fail{s3,1}((f+1),2)==0)</pre>		
744 745	<pre>System(s2).Element_Failure_Support(1,2)=Element_Fail{s3,1}((f+1),1); end</pre>		
746 747	end end		
748 749	# 3. Calculation of support length of all MPs for s4-1.No of Elements		
750	System(s4).Element_Failure_Support_Length(1,1)=(System(s4).Element_Failure_Support rt(1,2)-System(s4).Element_Failure_Support(1,1));		
751 752	end		
753	* Thengetting the start and the end of the modal values of each Element-Pailure 4 MP		
754	* 2. Start of modal		

Dec	ember 2, 2004 4:41:32 PM
755	for m2=1:No of Elements
756	for ff=1: (Col time-1)
757	if (Element Fail{m2,1}(ff,2) <max(element &="" &<="" (element="" fail{m="" fail{m2,1}(i,2)))="" td=""></max(element>
	2.1 ((ff+1).2)==max(Element Pail(m2.1)(1.2)))
758	System(m2),Element Failure Modal(1,1)=Element Fail{m2,1}(ff,1);
759	end
760	end
761	end
762	\$ 2. End of Support
763	for m3=1:No of Elements
764	for ff-1: (Col time-1)
765	if (Element Fail(m3,1)(ff,2) max(Element Fail(m3,1)(r,2))) & (Element Fail)
	m3,1}((ff+1),2) <max(element fail[m3,1](:,2)))<="" td=""></max(element>
766	System(m3).Element_Failure_Modal(1,2)=Element_Fail{m3,1}((ff+1),1);
767	end
768	end
769	end
770	# 3. Calculation of support length of all MPs
771	for m4-1/No_of_Elements
772	System(m4).Element_Failure_Modal_Length(1,1)=(System(m4).Element_Failure_Modal(1 &
	(2)-System(m4).Element_Failure_Modal(1,1));
773	end
774	
775	
776	# Build the sequence of Augmentation Sub-System step wise
777	<pre># Connection between sub_system is considered serial connection</pre>
778	
779	System_Fail_Information= System(1).Element_global_No(1,1) System(1).Element_Failure_ d
	Support_Length(1,1) System(1).Element_Failure_Modal_Length(1,1);
780	System(2).Element_global_No(1,1) System(2).Element_Failure_S 🕊
	upport_Length(1,1) System(2).Element_Failure_Modal_Length(1,1)
781	System(3).Element_global_No(1,1) System(3).Element_Failure_S 🛩
	upport_Length(1,1) System(3).Element_Failure_Modal_Length(1,1);
782	System(4).Element_global_No(1,1) System(4).Element_Failure_S 🖌
	upport_Length(1,1) System(4).Element_Failure_Modal_Length(1,1);
783	System(5).Element_global_No(1,1) System(5).Element_Failure_S 🕊
	upport_Length(1,1) System(5).Element_Failure_Modal_Length(1,1)
784	System(6).Element_global_No(1,1) System(6).Element_Failure_S 化
	upport_Length(1,1) System(6).Element_Failure_Modal_Length(1,1);
785	System(7).Element_global_No(1,1) System(7).Element_Failure_S 🕊
	upport_Length(1,1) System(7).Element_Failure_Mcdal_Length(1,1);
marc	Suprement (12) Planant slabal No(1 1) Supreme (12) Planant Pailure /

Dec	ember 2, 2004 4:41:32 PM
972 -	_Support_Length(1,1) System(13).Element_Failure_Modal_Length(1,1);
787	System(34).Element_global_No(1,1) System(34).Element_Failure 🖌
	_Support_Length(1,1) System(74).Element_Failure_Modal_Length(1,1),
788	System(42).Element_global_No(1,1) System(42).Element_Failure 🗸
	_Support_Length(1,1) System(42).Element_Failure_Modal_Length(1,1);
789	System (46).Element_global_No(1,1) System (46).Element_Failure 化
	_Support_Length(1,1) System(46).Element_Failure_Modal_Length(1,1);
790	System (58).Element_global_No(1,1) System (58).Element_Failure 化
	_Support_Length(1,1) System(58).Element_Failure_Modal_Length(1,1)/
791	System(60).Element_global_No(1,1) System(60).Element_Failure 🗸
	_Support_Length(1,1) System(60).Element_Failure_Modal_Length(1,1)
792	System(63).Element_global_No(1,1) System(63).Element_Failure 🖉
	_Support_Length(1,1) System(63).Element_Failure_Modal_Length(1,1);
793	System(68).Element_global_No(1,1) System(68).Element_Failure 🖌
	_Support_Length(1,1) System(68).Element_Failure_Modal_Length(1,1);
794	System(70).Element_global_No(1,1) System(70).Element_Failure 🗸
	_Support_Length(1,1) System(70).Element_Failure_Modal_Length(1,1)
795	System(75).Element_global_No(1,1) System(75).Element_Failure 🗸
	_Support_Length(1,1) System(75).Element_Failure_Modal_Length(1,1)],
796	
797	* Determine the controling MF, (first MF with max support, then MF with max
798	* modal)
799	
800	[Row_Fail,Col_Fail]=size(System_Fail_Information) /
801	
80.2	<pre>max_support=max(System_Fail_Information(:,2));</pre>
902	<pre>max_support_elements=find(System_Fail_Information(:,2)==max_support);</pre>
804	[Row_max_support_elements,Col_max_support_elements) = size (max_support_elements);
805	
806	if (Row_max_support_elements>1)
807	<pre>max_modal=max(System_Fail_Information(:,3));</pre>
808	<pre>max_modal_elements=find(System_Fail_Information(:,3)==max_modal);</pre>
809	[Row_max_modal_elements,Col_max_modal_elements]=size(max_modal_elements);
810	if (Row_max_modal_elements>1)
811	System_Fail=[System_Fail_Information(max_modal_elements(1,1),1) System_Fail_ 🖌
	Information(max_modal_elements(1,1),2) System_Fail_Information(max_modal_elements(1, *
	1),3)]/
912	end
313	System_Fail=[System_Fail_Information((max_modal_elements(1,1)==max_modal),1) Sys 🖌
	<pre>tem_Fail_Information((max_modal_elements(1,1)==max_modal),2) System_Fail_Information </pre>
	((max_modal_elements(1,1)==max_modal),3)];
814	end

C:\ Dec	Documents and Settings\Ibrahim\Desktop\Daily w\rmLH.m Page 24 ember 2, 2004 4:41:32 PM
815	57
816	System_Fail=[System_Fail_Information(max_support_elements(1,1),1) System_Fail_Inform & ation(max_support_elements(1,1),2) System_Fail_Information(max_support_elements(1,1) & ,3)];
817	
818	***************************************
819	***Step TEN-FOURuse the MF of the controling Element
82.0	System_Fail_MF=Element_Fail{System_Fail(1,1),1};
831	
822	<pre>[row_t,col_t]=size(System_Fail_MF);</pre>
833	
824	Resiltence_Index=0;
825	nominator=0;
826	denominator=0;
827	for t=1:(row_t-1)
828	nominator=nominator+(0.5*(System_Fail_MF(t+1,1)+System_Fail_MF(t,1)));
829	denominator=denominator+(System_Fail_NF(t,2)*(0.5*(System_Fail_MF(t+1,1)+System_*
	Fail_MF(t,1))));
830	end
831	Resilience_Index-nominator/denominator
83.2	

II-B ELGIN AREA PRIMARY WATER SUPPLY SYSTEM (LHPWSS)

```
C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmAE.m
                                                                                  Page 1
December 2, 2004
                                                                             4:54:54 PM
 1
    % Last Modified: Thursday, October 07, 2004 @5:45pm
 2
     x=-1:0.01:4;
 3
    time=0:0.1:200;
 4
    alpha=0:0.05:1;
 5 %%%......Step ONE.....Building Structure Array
 6 $$$it is an array that containes fields under each fieldthere are values or
 7
     ***strinngs
 A.
 9
         %determines the number of total elements in the system
10
         [No_of_Elements,Dummy01]=size(rowheaders);
11
1.2
         * Build the Structure array from the imported excel file.
13
         for n=1:No of Elements
14
                 System(n) = struct('Element_Name', rowheaders(n,1),'Element_ID',data(n,1 e
     :4) . . . .
15
                     'Element_global_No', data(n, 5), 'Element_Capacity', data(n, 6:8), ...
16
                     'Element Requirment', data(n,9:11), 'Element Norm Cap Tri', data(n,12:1 #
     4), 'Element Norm Req Tri', data(n, 15:17), ...
17
                     'Element Norm Cap Trap', data (n, 18:21), 'Element Norm Reg Trap', data (n 🖌
     ,22:25),...
18
                     'Element State TriMos', data(n, 26:28), 'Element State TrapMos', data(n, 🖌
     29:32),...
19
                     'Element Failure Tri', data(n, 33:35), 'Element Failure Trap', data(n, 36 🖉
     :39))/
20
         and
21
22
     *** ......Step TWO.....Deterimine the No. of Sub-Systems
23
     $8% by caclualting the maximum No. Value in the ID first two digits 44 Deterimine
24
     *** the No. of elements in each Sub-System by caclualting the maximum No. Value in
25
     *** the ID last two digits
26
27
             for sse=1:No of Elements
                                                 % sse stands for Sub-System Element
28
                 % Calculation of the Number of Sub-Systems
29
                 SubSystem (sse, 1) = [ [System (sse).Element ID(1, 1) System (sse).Element ID(1, 4
     2)]];
30
                 SubSystm Tenth(sse,1)=num2str(SubSystem{sse,1}(1,1));
31
                 SubSystm Units(sse, 1) = num2str(SubSystem{sse, 1}(1, 2));
32
                 SubSystem_String_Code=strcat(SubSystm_Tenth,SubSystm_Units);
33
                 SubSystem Numerical Code=str2num(SubSystem String Code);
34
                 Total SubSystems-max(SubSystem Numerical Code);
35
36
                 * Calculation of the Number of Elements in each Sub-System
```
C:\ Dec	Documents ember 2,	and Settings\Ibrahim\Desktop\Daily w\1 2004	:mAE.m	Page 2 4:54:54 PM
37		Element(sse,1)={[System(sse).Element ID(1,3) System	em(sse).El	ement ID(1,4) 🖌
	1};			1999 - 1999 -
38	999200	SubSystm Element Tenth(sse, 1)-num2str(Element(sse,	1)(1,1));	
39		SubSystm Element Units(sse,1)=num2str(Element(sse,	1)(1,2));	
40		Element_String_Code=strcat(SubSystm_Element_Tenth,	SubSystm_	Element Units 🖌
) :			
41		Element Numerical Code=str2num(Element String Code	ə);	
42		end		
43				
44	System_Nu	meric_ID_Array=[SubSystem_Numerical_Code_Element_Nume	arical_Cod	ie];
45				
46	* Creatin	g sero matrix to store information on number of eleme	ents in ea	oh subsystem
47	Informati	on_Array=reros(Total_SubSystems,2);		
48				
49	counter-1	15		
50	for s	se2=1:(No_of_Elements-1)		
51	I	nformation_Array(counter,1)=counter;		
52	I	nformation_Array(counter,2)=System_Numeric_ID_Array(s	sse2,2);	
53				
54		<pre>if (System_Numeric_ID_Array((sse2+1),2))<(System_Numeric_ID_Array((sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(System_Numeric_ID_Array(sse2+1),2))<(</pre>	Americ_ID	_Arrsy(sse2,2 ¢
	3.5			
55		counter-counter+1;		
56		end		
57				
58		if counterTotal_SubSystems		
59		Information_Array(counter,2)=System_Numeric_ID_	_Array(sse	2+1,2);
60		end		
61	end			
62				
63	********	***************************************	******	
64	***	Step THREEMake Puzzy MF usable in Augmentat	tion ð	
65	\$\$\$ Using	the Triangula NP in the system Matrix	8	
66		for i=1:No_of_Blements	8	
67	20	a=System(1).Blement_State_TriMos(1,1);	8	
68	8	b=System(i).Element_State_TriMos(1,2);	÷.	
69	. 4	c=System(i).Element_State_TriMcs(1,3);	8	
70	2	<pre>Element_MF(1,1)={(triangular(alpha,a,b,d))};</pre>	8	
71	8 8	nd	8	
72	********	***************************************	188888	
73	********	***************************************	688888	
74	***Step THREE Make Fuzzy MF usable in Augmentation *			
75	*** Using the Trapisoidal MF in the system Matrix *			

C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmAE.m Page 3 December 2, 2004 4:54:54 PM 76 for i=1:No of Elements * 77 a=System(i).Element_State TrapMos(1,1); * 78 b-System(1).Element_State_TrapMos(1,2)/ ÷. 79 c=System(i).Element State TrapMos(1,3); 4 80 d=System(i).Element_State_TrapMos(1,4); * 81 Element MF(1,1)={[trapizoidal(alpha,a,b,c,d)]}; ÷ 82 and 83 84 85 86 ### using the SUMMATION operator for both 87 ** NOTE: all redundant elements or parallel elelements will be added to the 88 ** first element in the group (all values will be incorporated in the MF 89 %% value of the first element) 90 statessessatessessatesses Redundant Elementssessatessessatessessatesses 91 92 %1. Intake Sub-System 93 % I Redundant elements 94 8 03----04 95 Element MF{3,1}=fuzzymath(alpha,Element MF{3,1},Element MF{4,1},'sum'); 2 96 \$2. Low Lifting Sub-System 97 # 3 Redundant elements 98 25----39, 27----40 & 29----42 99 Element MF{25,1}=fuzzymath(alpha,Element MF{25,1},Element MF{39,1},'sum'); 100 Element MF{27,1}-fuzzymath(alpha,Element MF{27,1},Element MF{40,1}, "sum"); 101 Element MF{29,1}-fuzzymath(alpha,Element NF{29,1},Element MF{41,1},'sum'); 102 103 \$3. Palsh Mixing Sub-System 104 \$PAC, Alum, and Polymer Sub system 105 * there is no redundancy in any of the elements 106 107 \$4. Flocculation Sub-System 108 * there is no redundancy in any of the elements 109 110 \$5. Sedimentation Sub-System 111 % 1 Redundant elements 112 \$ 51----58 113 Element_MF{51,1}=fuzzymath(alpha,Element_MF{51,1},Element_MF{58,1},'sum'); 114 115 \$6. Filtering Sub-System 116 * there is no redundancy in any of the elements 16

Dec	ember 2, 2004 4154:54 PM		
117			
118	 High Lifting Sub-System 		
119	% there is no redundancy in any of the elements		
120			
121	 Conveyance Sub-System 		
122	* there is no redundancy in any of the elements		
123			
124	\$9. Storage Sub-System		
125	<pre>% there is no redundancy in any of the elements</pre>		
126			
127	\$		
128	****************************** Parallel Elements************************************		
	8888		
129	%1. Intake Sub-System		
130	there is no parallel elements		
131	S manufal exception and a metallicity of the second secon second second sec		
132	\$2. Low Lifting Sub-System		
133	\$ 4 Paralel elements		
134	\$ 09-10-11-12 *		
135	Element MP(9,1)=fuzzymath(alpha,Element MP(9,1),Element MP(10,1),'sum');		
136	Element MF(9,1)=fuzzvmath(alpha,Element MF(9,1),Element MF(11,1),'sum');		
137	Element MF(9,1)=fuzzymath(alpha,Element MF(9,1),Element MF(12,1),'sum');		
138			
139	%). Palsh Mixing Sub-System		
140	*PAC, Alum, and Polymer Sub system		
141	* 14-15		
142	\$ 16-17		
143	* 19-20-21-22		
144	* 23-24		
145	\$ 27-28-29-30		
146	\$ 31-32		
147	\$ 35-36-37-38		
148	Element MF{14,1}=fuzzymath(alpha,Element MF{14,1},Element MF{15,1},'sum');		
149	Element MF{16,1}=fuzzymath(alpha,Element MF{16,1},Element MF{17,1},'sum');		
150	Element MF(19,1)=fuzzymath(alpha,Element NF(19,1).Element NF(20,1).'sum'):		
151	Element MF(19,1)=fuzzvmath(alpha,Element NF(19,1),Element NF(21,1),'sum')		
152	Element_MF{19,1}=fuzzymath(alpha,Element_MF{19,1},Element_MF{22,1},'sum');		
153	Element MF{23,1]=fuzzymath(alpha,Element MF{23,1},Element MF{24,1},'sum');		
153	Element_MF{23,1}=fuzzymath(alpha,Element_MF{23,1},Element_MF{24,1},'sum');		

C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmAE.m Page 4 December 2, 2004 4:54:54 PM

```
C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmAE.m
                                                                                  Page 5
 December 2, 2004
                                                                             4:54:54 PM
154
      Element MF{27,1}=fuzzymath(alpha,Element MF{27,1},Element MF{28,1},'sum');
155
      Element_MP{27,1}=fuzzymath(alpha,Element_MF{27,1},Element_MF{29,1},'sum');
156
      Element_MP{27,1}-fuzzymath(alpha,Element_MP{27,1},Element_MP{30,1},'sum');
157
      Element MP{31,1}=fuzzymath(alpha,Element NP{31,1},Element MP{32,1},'sum');
158
      Element_MF{35,1}=fuzzymath(alpha,Element_MF{35,1},Element_MF{36,1},'sum');
159
      Element MP{35,1}=fuzzymath(alpha,Element MP{35,1},Element MP{37,1},'sum');
160
      Element_MP{35,1}=fuzzymath(alpha,Element_MP{35,1},Element_MP{38,1},'sum');
161
162
                           $4. Plocculation Sub-System
163
                                  % 4 Paraleï elemente
154
                                  $ 42-43-44-45
165
      Element MF{42,1}=fuzzymath(alpha,Element MF{42,1},Element MF{43,1},'sum');
166
      Element MF{42,1]=fuzzymath(alpha, Element MF{42,1], Element MF{44,1], 'sum');
167
      Element_MF{42,1}=fuzzymath(alpha,Element_NF{42,1},Element_NF{45,1},'sum');
168
169
                           $5. Sedimentation Sub-System
170
                              % 2 Paralel groups of elements
171
                               8 46-47
172
                               $ 49-50
173
174
      Element MF{46,1}=fuzzymath(alpha,Element MF{46,1},Element MF{47,1},'sum');
175
      Element_MF{49,1}=fuzzymath(alpha,Element_MF{49,1},Element_MF{50,1},'sum');
176
177
                           %6. Piltering Sub-System
178
                                  # 4 Paralel elements
179
                                  $ 52-53-54-55
180
      Element MF{52,1}-fuzzymath(alpha,Element NF{52,1},Element MF{53,1},'sum');
181
       Element MP{52,1}-fuzzymath(alpha,Element NP{52,1},Element MP{54,1},'sum');
182
      Element MF{52,1}-fuzzymath(alpha,Element MF{52,1},Element MF{55,1},'sum');
183
184
                           $7. High Lifting Sub-System
185
                                  % 4 Paralel elements
186
                                  $ 60-61-62-63
      Element MP{60,1}-fuzzymath(alpha,Element NP{60,1},Element NP{61,1},'sum');
187
188
      Element MF{69,1}=fuzzymath(alpha,Element NF{60,1},Element NF{62,1},'sum');
189
      Element_MP{60,1}=fuzzymath(alpha,Element_MP{60,1},Element_MP{63,1},'sum');
190
191
                           $8. Conveyance Sub-System
192
                                  % there is no parallel elements
193
194
                           $9. Storage Sub-System
195
                                  % 2 Paralel elements
```

Dec	ember 2, 2004 4:54:54 PM		
196	\$ 65-66		
197	Element MF{65,1}=fuzzvmath(alpha.Element MF{65,1}.Element MF{66,1}.'sun');		
198	***************************************		
	8888		
199	**************************************		
	3333		
200			
201	% Build the sequence of Augmentation Sub-System step wise		
202	* Connection between sub system is considered serial connection		
203	17.8		
204	 Intake Sub-System 		
205	≇ 1 serial 2 serial 3≠		
206	<pre>\$ star element means it has a rdundant element</pre>		
207	System_State=fuzzylogmath(x,alpha,Element_MF{1,1},Element_MF{2,1},'minimum');		
208	System_State=fuzzylogmath(x,alpha,System_State,Element_MF(3,1),'minimum');		
209			
210	 Low Lifting Sub-System 		
211	\$ 5 serial 7 serial (9#) serial 13		
212	System_State=fuszylogmath(x,alpha,System_State,Element_MP{5,1},'minimum');		
213	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{7,1},'minimum');		
214	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{9,1},'minimum');		
215	System_State=fuzzylogmath(x,alpha,System_State,Element_MF(13,1},'minimum');		
216			
217	 Falsh Mixing Sub-System 		
218	%PAC, Alum, and Polymer Sub_system		
219	# (14#) serial (16#) serial 18 serial (19#) serial (23) serial		
	ial		
220	# 25* serial 26 serial (27#) serial (31#) serial 33 serial		
	34 serial (35#)		
221	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{14,1},'minimum');		
222	<pre>System_State=fuzzylogmath(x,alpha,System_State,Element_MF{16,1},'minimum');</pre>		
223	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{18,1},'minimum');		
224	<pre>System_State=fuzzylogmath(x,alpha,System_State,Element_MF{19,1},'minimum');</pre>		
225	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{23,1},'minimum');		
226	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{25,1},'minimum');		
227	<pre>System_State=fuzzylogmath(x,alpha,System_State,Element_MF{26,1}, 'minimum');</pre>		
228	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{27,1},'minimum');		
229	<pre>System_State=fuzzylogmath(x,alpha,System_State,Element_MF{31,1},'minimum');</pre>		
230	<pre>System_State=fuzzylogmath(x,alpha,System_State,Element_MF{33,1},'minimum');</pre>		
231	System_State=fuzzylogmath(x,alpha,System_State,Element_MF(34,1},'minimum'))		
232	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{35,1},'minimum');		
233			

234	#4. Plocculation Sub-System	
235	% pp serials only previous system-state	
236	Custam State_furrylogmath/y alpha Sustam State Element ME(42 1) Iminimum();	
237	ofores or one for of the description of the second or one of the fact of the second of	
23.8	 Sedimentation Sub-System 	
23.9	% (46#) serial 48 serial (50#) serial 51	
240	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{46,1},'minimum');	
241	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{48,1}, 'minimum');	
24.2	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{50,1},'minimum');	
243	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{51,1},'minimum');	
24.4		
245	 Filtering Sub-System 	
246	<pre>\$ (52#) serial 56 serial 57</pre>	
247	System_State=fuzzylogmath(x,alpha,System_State,Element_MF(52,1},'minimum');	
248	System_State=fuzzylogmath(x,alpha,System_State,Element_MF(56,1},'minimum');	
249	System_State=fuzzylogmath(x,alpha,System_State,Element_MF(57,1},'minimum');	
250		
251		
252	\$7. High Lifting Sub-System	
253	1 59 serial (60#)	
254	System_State=fuzsylogmath(x,alpha,System_State,Element_MF{59,1},'minimum');	
255	System_State=fuzzylogmath(x,alpha,System_State,Element_MF(60,1},'minimum');	
256		
257	 Conveyance Sub-System 	
258	no serials, only previous system-state	
259	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{64,1},'minimum');	
260		
261	 Storage Sub-System 	
262	no serials, only previous system-state	
263	System_State=fuzzylogmath(x,alpha,System_State,Element_MF{65,1},'minimum');	
264		
265	[Row_x,Col_x]=size(x); * basic information about x-values (x-axis)	
266	No_x_steps=Col_x-1;	
267	x step=max(x)/No_x steps;	
269	***************************************	
269	* keeping the original System State MF with zeros for farther use in	
270	* calcuation of the overlap area	
271	System_State_xwise=System_State;	
272	***************************************	
273		
274	System_State=[System_State(find(System_State(;,2)>0),1) System_State(find(System_State) te(:,2)>0),2)];	

C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmAE.m Page 7 December 2, 2004 4:54:54 PM

Dec	ember 2, 2004 4:54:54 PM
275	[dumm row dumm coll-size(System State);
276	System State= (System State(1,1) -x step) 0:System State: (System State(dumm row.1) +x 🖌
3020	step) 0],
277	
278	************System State Areas**********************************
279	* Determine the maximum alpha value in teh system-state matrix.
280	Maximum alpha System State=max(System State(:,2));
281	
282	* based on the No. of rows, a loop will search for the row index of the
283	≱ maximum alpha value
284	[row_System_State.column_System_State]=size(System_State);
285	max index System State=0;
286	for t=1:row_System_State
287	max_index_System_State=max_index_System_State+1;
288	if System_State(t,2)==Maximum_alpha_System_State
28.9	break;
290	end
291	end
293	
293	% the range from the maximum value to Zero will be divided for high
294	<pre>% resolution alpha.</pre>
295	alpha_range_System_State=Maximum_alpha_System_State;
296	step_no_System_State=100;
297	<pre>step_System_State=Maximum_alpha_System_State/step_no_System_State;</pre>
298	
299	<pre>%Interpolation</pre>
300	%Left-Hand Interpolation
301	Left_System_State=(interpl(System_State(1:max_index_System_State,2),Syst 🖌
	em_State(1:max_index_System_State,1),0:step_System_State:Maximum_alpha_System_State) 化
	*) ;
302	System_State_L=[Left_System_State (0:step_System_State:Maxinum_alpha_Sys 🖌
	tem_State)'];
303	<pre>%Right-Hand Interpolation</pre>
304	Right_System_State= (interp1 (System_State(max_index_System_State:row_Syst 🖌
	em_State,2),System_State(max_index_System_State:row_System_State,1),O:step_System_St 🖌
	ate:Maximum_alpha_System_State)');
305	System_State_R=[Right_System_State (0:step_System_State:Maximum_alpha_Sy 🖌
	<pre>sten_State) '];</pre>
306	
307	%Area-Calculation
308	strip_Area_System_State=0;
309	for tt=1: (step_no_System_State-1)

```
C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmAE.m
                                                                       Page 9
 December 2, 2004
                                                                   4:54:54 PM
310
            strip_Area_System_State=strip_Area_System_State+(0.5*((System_State_R(tt,1)- 化
     System State L(tt,1))+(System State R((tt+1),1)-System State L((tt+1),1)))*step Syst &
     em_State);
311
         end
312
         Area_System_State=strip_Area_System_State+(0.5*(System_State_R((step_no_System_S *
     tate+1).1)-System State L((step no System State+1).1))*step System State);
313
314
         Weighted Area-Calculation
315
         weighted strip Area System State=0;
316
         for ttt=1:(step_no_System_State-1)
317
            weighted_strip_Area_System_State=weighted_strip_Area_System_State+((0.5*((Sy .
      sten State R(ttt,1)-System State L(ttt,1))+(System State R((ttt+1),1)-System State L 4
     ((ttt+1),1)))*step System State)*((System State L(ttt,2)+System State L((ttt+1),2))* 4
     0.5))/
318
         end
319
         Weighted Area_System_State=weighted_strip_Area_System_State+((0.5*(System_State_ <
     R((step no System State+1),1)-System State L((step no System State+1),1))*step Syste 🖌
     m State)*System State L(step no System State,2));
320
         321
322
     323 ### ......Step FIVE.....Define the Acceptable level(s) of performance
                                                                      ÷
324
     ### using trapizoidal MF as (a,b,c,d) where c-d
                                                                      $
325 No of Levels=3;
                                                                      ÷
326 m1=-0.5; m2=4.5; m3=5; m4=5;
                                                                      1
327 Level ly=trapmf(x,[m1 m2 m3 m4]);
328 Level 1-[x' Level 1y'];
329
     LR Level 1= (m1*m2) / (m2-m1);
330
331 n1=0.2; n2=1; n3=5; n4=5;
                                                                     *
332 Level 2y=trapmf(x,[n1 n2 n3 n4]);
333 Level_2=[x' Level_2y'];
334 LR Level 2=(n1*n2)/(n2-n1);
335
336 q1=-1.0; q2=5; q3=5; q4=5;
                                                                     1
337 Level 3y=trapmf(x,[q1 q2 q3 q4]);
338 Level 3=[x' Level 3y'];
339 LR_Level_3=(q1*q2)/(q2-q1);
340
341 LR=[LR_Level_1;LR_Level_2;LR_Level_3];
342
```

```
C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmAE.m
                                                                       Page 10
 December 2, 2004
                                                                    4:54:54 PM
     344
345
     *** .....Step SIX.....Calculation of the overlap Matrix by taking the minimum
346
    *** of the state function and each level
347 OverLap Level 1y=min(Level 1(:,2),System State xwise(:,2));
348 OverLap_Level_1=[x' OverLap_Level_1y];
349
350 OverLap_Level_2y=min(Level_2(:,2),System_State_xwise(:,2));
351 OverLap_Level_2=[x' OverLap_Level_2y];
352
353 OverLap_Level_3y=min(Level_3(:,2),System_State_xwise(:,2));
354
    OverLap_Level_3=[x' OverLap_Level_3y];
355
356
         % take-off the seros from both ends of the overlap area
357
358 Area Level 1=OverLap Level 1((find(OverLap Level 1(:,2)>0)),:);
359
     Area_Level_1_start=[min(OverLap_Level_1((find(OverLap_Level_1(:,2)>0)),1)) -x_step 0] <
     12
360
     Area Level 1 end-[max(OverLap Level 1((find(OverLap Level 1(:,2)>0)),1))+x step 0];
361
     Area Level 1= [Area Level 1 start; Area Level 1; Area Level 1 end];
362
363 Area Level 2=OverLap Level 2((find(OverLap Level 2(:,2)>0)),:);
364 Area_Level_2_start=[min(0verLap_Level_2((find(0verLap_Level_2(:,2)>0)),1))-x_step 0] 4
365
     Area Level 2 end-[max(OverLap Level 2((find(OverLap Level 2(:,2)>0)),1))+x step 0];
366
     Area Level 2-[Area Level 2 start/Area Level 2/Area Level 2 end]/
367
369
369
     Area Level 3=OverLap Level 3((find(OverLap Level 3(:,2)>0)),:))
370
     Area Level 3 start=[min(OverLap Level 3((find(OverLap Level 3(:,2)>0)),1)) -x step 0] @
371 Area_Level_3_end=[max(OverLap_Level_3((find(OverLap_Level_3(:,2)>0)),1))+x_step 0];
372 Area_Level_3=[Area_Level_3_start;Area_Level_3;Area_Level_3_end];
373
375 ### .....Step SEVEN....Calculation of the overlap area & weighted area by inte @
     prolation over alpha cuts
376 $8% for each overlap matrix
377
378
        379
         * Determine the maximum alpha value in teh ovelap matrix.
380
         Maximum alpha Area Level 1=max(Area Level 1(:,2));
381
```

```
C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmAE.m
                                                                                 Page 11
 December 2, 2004
                                                                             4:54:54 PM
382
           * based on the No. of rows, a loop will search for the row index of the
383
           % maximum alpha value
384
          [row_Area_Level_1, column_Area_Level_1]=size(Area_Level_1))
385
          max index Area Level 1=0;
386
          for ii=1:row Area Level 1
387
              max index Area Level 1-max index Area Level 1+1;
388
              if Area_Level_1(ii,2)==Maximun_alpha_Area_Level_1
389
                   break:
390
              end
391
          end
392
          % the range from the maximum value to Zero will be divided for high
393
394
          * resolution alpha.
395
          alpha_range_Area_Level_1=Maximum_alpha_Area_Level_1/
396
          step no Area Level 1-100;
397
          step_Area_Level_1=Maximum_alpha_Area_Level_1/step_no_Area_Level_1;
398
399
          %Interpolation
400
              %Left-Hand Interpolation
401
                  Left Area Level 1-(interpl(Area Level 1(1:max index Area Level 1,2),Area 🖌
      Level 1(1:max index Area Level 1,1),0:step Area Level 1:Maximum alpha Area Level 1) 🕊
      *)1
402
                  Area Level 1 L-[Left Area Level 1 (0:step Area Level 1:Maximum alpha Are 🖉
      a Level 1)'];
403
              *Right-Hand Interpolation
404
                  Right Area Level 1= (interpl(Area Level 1(max index Area Level 1:row Area 🖉
      Level 1,2) Area Level 1(max index Area Level 1:row Area Level 1,1),0:step Area Leve &
      1 1:Maximum alpha Area Level 1) *);
405
                  Area Level 1 R=[Right Area Level 1 (0:step Area Level 1:Maximum alpha Ar @
      ea Level 1)';
406
407
          %Area-Calculation
408
          strip Area Area Level 1=0;
409
          for jj-1: (step no Area Level 1-1)
              strip_Arss_Ares_Lavel_1=strip_Ares_Ares_Lavel_1+(0.5*((Ares_Lavel_1_R(jj,1)- *
410
      Area_Level_1_L(jj,1))+(Area_Level_1_R((jj+1),1)-Area_Level_1_L((jj+1),1))*step_Area &
      Level 1);
411
          end
412
          Area_Area_Level_1=strip_Area_Area_Level_1+(0.5*(Area_Level_1_R((step_no_Area_Lev w
      el i+1),1)-Area Level 1 L((step no Area Level 1+1),1))*step Area Level 1);
413
414
          Weighted Area-Calculation
```

```
C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmAE.m
                                                                               Page 12
 December 2, 2004
                                                                           4:54:54 PM
415
          weighted strip Area Area Level 1=0;
416
          for kk=1: (step no Area Level 1-1)
417
              weighted_strip_Area_Area_Level_1=weighted_strip_Area_Area_Level_1+((0.5*((Ar 🖌
      ea Level 1 R(kk,1)-Area Level 1 L(kk,1))+(Area Level 1 R((kk+1),1)-Area Level 1 L((k 🖌
      k+1),1))*step_Area_Level_1)*((Area_Level_1_L(kk,2)+Area_Level_1_L((kk+1),2))*0.5));
418
          and
419
          Weighted_Area_Area_Level_1=weighted_strip_Area_Area_Level_1+((0.5*(Area_Level_1_ <
      R((step_no_Area_Level_1+1),1)-Area_Level_1_L((step_no_Area_Level_1+1),1))*step_Area_ #
      Level_1)*Area_Level_1_L(step_no_Area_Level_1,2));
420
          $$$$$$$$$$.....Level ONE....$$$$$$$$$$$$$$$$$$$$$$$$$$
421
          422
423
          % Determine the maximum alpha value in teh ovelap matrix.
424
          Maximum_alpha_Area_Level_2=max(Area_Level_2(:,2));
425
426
          * based on the No. of rows, a loop will search for the row index of the
427
          * maximum alpha value
428
          [row Area Level 2, column Area Level 2]=size(Area Level 2);
429
          max index Area Level 2+0;
430
          for iii-1:row Area Level 2
431
              max index Area Level 2=max index Area Level 2+1;
432
              if Area Level 2(111,2) -- Maximum alpha Area Level 2
433
                   break:
434
              end
435
          and
436
437
          * the range from the maximum value to Zero will be divided for high
438
          * resolution alpha.
439
          alpha range Area Level 2-Maximum alpha Area Level 2;
44.0
          step no Area Level 2=100;
441
          step_Area_Level_2=Maximum_alpha_Area_Level_2/step_no_Area_Level_2/
44.2
443
          *Interpolation
444
              %Left-Hand Interpolation
445
                  Left Area Level 2-(interp1(Area Level 2(1:max index Area Level 2,2), Area @
       _Level_2(1:max_index_Area_Level_2,1),0:step_Area_Level_2:Maximum_alpha_Area_Level_2) 🖌
      115
446
                  Area_Level_2_L=[Left_Area_Level_2 (0:step_Area_Level_2:Maxinum_alpha_Are «
      a Level 2)'];
447
              *Right-Hand Interpolation
448
                  Right Area Level 2=(interp1(Area Level 2(max index Area Level 2:row Area *
      _Level_2,2), Area_Level_2(max_index_Area_Level_2:row_Area_Level_2,1), 0:step_Area_Leve 🖌
```

```
C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmAE.m
                                                                                 Page 13
 December 2, 2004
                                                                              4:54:54 PM
       1 2:Maximum alpha Area Level 2)'):
449
                  Area Level 2 R=[Right Area Level 2 (0:step Area Level 2:Maximum alpha Ar 🖌
      ea_Level_2)'| /
450
451
          *Area-Calculation
452
          strip Area Area Level 2=0;
453
          for jjj=1:(step_no_Area_Level_2-1)
454
              strip Area_Area_Level_2=strip Area_Area_Level_2+(0.5*((Area_Level_2_R(j)j,1) #
      -Area_Level_2_L(jjj,1))+(Area_Level_2_R((jjj+1),1)-Area_Level_2_L((jjj+1),1)))*step_
      Area_Level_2);
455
          ond
456
          Area Area Level 2-strip Area Area Level 2+(0.5*(Area Level 2 R((step no Area Lev d
      el 2+1),1)-Area Level 2 L((step no Area Level 2+1),1))*step Area Level 2);
457
458
          Weighted Area-Calculation
459
          weighted_strip_Area_Area_Level_2=0;
460
          for kkk=1: (step no Area Level 2-1)
461
              weighted strip Area Area Level 2-weighted strip Area Area Level 2+((0.5*((Ar 4
       ea Level 2 R(kkk,1)-Area Level 2 L(kkk,1))+(Area Level 2 R((kkk+1),1)-Area Level 2 L 🖌
      ((kkk+1),1)))*step Area Level 2)*((Area Level 2 L(kkk,2)+Area Level 2 L((kkk+1),2))* •
      0.5));
462
          end
463
          Weighted Area Area Level 2-weighted strip Area Area Level 2+((0.5*(Area Level 2 🖌
      R((step no Area Level 2+1),1)-Area Level 2 L((step no Area Level 2+1),1))*step Area 🕊
       Level 2) * Area Level 2 L(step no Area Level 2,2));
464
           ***************.....Level TWO....*****************************
465
466
           ************.....Level THREE...*****************************
467
          * Determine the maximum alpha value in teh ovelap matrix.
468
          Maximum alpha Area Level 3=max(Area Level 3(;,2));
469
470
          * based on the No. of rows, a loop will search for the row index of the
471
          * maximum alpha value
472
          [row Area Level 3, column Area Level 3]=size(Area Level 3);
473
          max index Area Level 3=0;
474
          for iiii=1:row Area Level 3
475
              max index Area Level 3-max index Area Level 3+1;
476
              if Area_Level_3(iiii,2)==Maxinum_alpha_Area_Level_3
477
                   break;
478
              end
479
          end
480
```

Dec	2004 4154.54 PM			
481	* the range from the maximum value to Serc will be divided for high			
182	<pre>% resolution alpha.</pre>			
83	alpha_range_Area_Level_3=Maximum_alpha_Area_Level_3;			
84	step_no_Area_Level_3=100;			
85	step_Area_Level_3=Maximum_alpha_Area_Level_3/step_no_Area_Level_3;			
86				
87	*Interpolation			
88	%Left-Hand Interpolation			
89	Left_Area_Level_3~(interpl(Area_Level_3(1:nax_index_Area_Level_3,2),Area)			
	_Level_3(1:max_index_Area_Level_3,1),0:step_Area_Level_3:Maximum_alpha_Area_Level_3) (
	$MT_{\rm eff}$ and a set with the set of which we have been been deally $MT_{\rm eff}$			
90	Area_Level_3_L=[Left_Area_Level_3 (0:step_Area_Level_3:Maximum_alpha_Are +			
	a_Level_3)'];			
91	%Right-Hand Interpolation			
92	Right_Area_Level_3=(interp1(Area_Level_3(max_index_Area_Level_3:row_Area)			
	_Level_3,2),Area_Level_3(max_index_Area_Level_3:row_Area_Level_3,1),0:step_Area_Leve			
	1_3:Maximum_alpha_Area_Level_3)');			
93	Area_Level_3_R=[Right_Area_Level_3 (0:step_Area_Level_3:Maximum_alpha_Ar (
	ea_Level_3)';			
94				
95	%Area-Calculation			
96	strip_Area_Area_Level_3=0;			
97	for jjjj-1:(step_no_Area_Level_3-1)			
98	strip_Area_Area_Level_3=strip_Area_Area_Level_3+(0.5*((Area_Level_3_R()))),1 +			
)-Area_Level_3_L(jjjj,1))+(Area_Level_3_R((jjjj+1),1)-Area_Level_3_L((jjjj+1),1)))*s			
	tep_Area_Level_3);			
99	end			
00	Area_Area_Level_3=strip_Area_Area_Level_3+(0.5*(Area_Level_3_R((step_no_Area_Lev +			
	el_3+1),1)-Area_Level_3_L((step_no_Area_Level_3+1),1))*step_Area_Level_3);			
01				
02	Weighted Area-Calculation			
03	weighted_strip_Area_Area_Level_3=0;			
04	for kkkk=1:(step_no_Area_Level_3-1)			
05	weighted_strip_Area_Area_Level_3=weighted_strip_Area_Area_Level_3+((0.5*((Ar			
	es_Lavel_3_R(kkkk,1) Ares_Level_3_L(kkkk,1)) + (Ares_Level_3_R((kkkk+1),1) - Ares_Level_4)			
	3_L((kkkk+1),1)))*step_Area_Level_3)*((Area_Level_3_L(kkkk,2)+Area_Level_3_L((kkkk+1))))			
),2))*0.5));			
0.6	end			
07	Weighted_Area_Area_Level_3=weighted_strip_Area_Area_Level_3+((0.5*(Area_Level_3_)			
	R((step no Area Level 3+1),1)-Area Level 3 L((step no Area Level 3+1),1))*step Area			

C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmAE.m Page 15 December 2, 2004 4:54:54 PM 509 510 Area=[Area Area Level 1;Area Area Level 2;Area Area Level 3]; 511 Max_Level=find(Area==max(Area)); 513 ***Step EIGHT Reliability measure Calculation 514 CM Level 1-Weighted Area Area Level 1/Weighted Area System State; 515 CM_Level_2=Weighted_Area_Area_Level_2/Weighted_Area_System_State; 516 CM_Level_3=Weighted_Area_Area_Level_3/Weighted_Area_System_State; 517 CM-[CM Level 1;CM Level 2;CM Level 3]; 518 519 Reliability Index=(max(CN)*LR(Max Level,1))/max(LR) 522 Robustness_Index_1_2=1/(CM_Level_1-CM_Level_2) 523 Robustness Index 1 3=1/(CM Level 1-CM Level 3) 524 Robustness_Index_2_3=1/(CM_Level_2-CM_Level_3) 527 528 529 ********************** 530 ************ 531 \$\$\$\$\$ 53.2 \$\$\$ 533 88 534 * 535 88 536 \$88 537 8888888 538 *********** 539 ************************** 543 ###Step TEN....Reglience measure Calculation 544 547 *** Using the Triangula NF in the system Matrix * 548 * for I=1:No of Elements * 549 8 aa=System(i).Element Pailure Tri(1,1); * 550 8 bb=System(i).Element Pailure Tri(1,2); *

	100.000	AND THE STOCK AND AND A VEHICLE WAS ADDRESS OF A DREAM AND A DR	300	
551	8	cc=System(i).Element_Pailure_Tri(1,3);	*	
552	*	<pre>Blement_Fail(i,1)={{triangular(alpha,aa,bb,cc)}};</pre>	8	
553	* •	end	8	
554	***************************************			
555	*******		***	
556	***	Step TEN-TWOMake Puzzy MP usable in Augmentation	8	
557	*** Using	g the Trapizoidal MP in the system Matrix	*	
558		for i=1:No_of_Elements	8	
559		aa=System(i).Element_Failure_Trap(1,1);	5	
560		bb=System(i).Element_Failure_Trap(1,2);	8	
561		cc=System(i)_Element_Failure_Trap(1,3);	*	
562		dd=System(i).Element_Failure_Trap(1,4);	8	
563		Element_Fail(i,1)=[[trapizoidal(alpha,aa,bb,cc,dd)]];	8	
564		end	3	
565	*******	***************************************	***	
000	222.22		1 - 24 C - N	
567	***	Step TEN-THKEEAugmenting the paralel and redunda	nc elemencs	
568	444 0810	I the MAXIMUM operator for Doth	4.4	
559	** NOTE:	all redundant elements or parallel elelements will be added	CO CRE	
570	<pre>## IIrst element in the group (all values will be incorporated in the MF</pre>		ne ar	
571	44 Value	of the first element)		
572		LEESENSESSESSES Davindant Diamontassessessessessesses		
574	30034935	At Intake Cub. Custem		
575		8 1 Dadundant alamant		
576		* 0304		
577	Elenant I	aill? 11-furrylogmath(time alpha Element Faill? 1) Element	Pail/4 1 Pmaxim	
232	11m ¹)+	arr(still-court communication (arrange, arrange, arrange, arrange)		
578				
579				
580		 Low Lifting Sub-System 		
581		% J Redundant element		
582		\$ 2539, 2740, 2941		
583	Element_Fail{25,1}=fuszylogmath(time,alpha,Element_Fail{25,1},Element_Fail{39,1},'m		nt_Fail{39,1},'ma 4	
584	Element_	<pre>%ail{27,1}=fuzzylogmath(time,alpha,Element_Fail{27,1},Eleme</pre>	nt_Fail{40,1},'ma	
585	Elenent 1	ail[29,1]=fuzzylogmath(time.alpha.Element Fail[29,1].Eleme	nt Fail{41,1}.'ma	
	<pre>ximum*);</pre>			
586				
		and Relate Manual Real Reaction		

Dec	ember 2, 2004	4:54:54 PM
589		% there is no redundancy in any of the elements
590		ne i la secola de la come de la comencia en entre en comencia de la comencia de la comencia de la comencia de c
591	¥d.	Plocoulation Sub-System
592		* there is no redundancy in any of the elements
593		
594	85.	Sedimentation Sub-System
595		% 3 Redundant element
596		\$ 5258
597	<pre>Element_Fail(51,1}=fussyl ximum*);</pre>	ogmath(time,alpha,Element_Fail{51,1},Element_Fail{58,1},'ma
59.8		
599	\$6.	Filtering Sub-System
600		* there is no redundancy in any of the elements
601		
602	37.	High Lifting Sub-System
603		% there is no redundancy in any of the elements
604		
605	₹8.	Conveyance Sub-System
606		there is no redundancy in any of the elements
607		
608	89.	Storage Sub-System
609		% there is no redundancy in any of the elements
610		
611	***********************	***************************************

612	**********************	Parallel Elements************************************

613	ðī.	Intake Sub-System
614		* there is no parallel elements
615		
516	\$2.	Low Lifting Sub-System
617		% 4 Paralel elements
618		\$ 09-10-11-12
619	<pre>Element_Fail{9,1}=fuzzylogmath(time,alpha,Element_Fail{9,1},Element_Fail{10,1},'maxi mim');</pre>	
620	<pre>Element_Fail{9,1}=fuzzylogmath(time,alpha,Element_Fail{9,1},Element_Fail{11,1},'max mum');</pre>	
621	<pre>Element_Fail(9,1)=fuzzylogmath(time,alpha,Blement_Fail(9,1),Element_Fail(12,1),'maxi mum');</pre>	
622		
523	#3.	Palsh Mixing Sub-System
		2030 Jun and Polymor Sub greaton

C:\ Dec	Documents and Settings\Ibrahim\Desktop\Daily w\rmAE.m Page 18 ember 2 2004 4:54:54 DN
625	
626	0 17-13 8 76 17
617	* 10-27 PT
620	* 12-20-21-22 * 22-24
629	a 20-24 8-27,28,24,20
630	* 31.32
631	\$ 36_36_37_38
632	<pre>Element_Fail{14,1}=fuzzylogmath(time,alpha,Element_Fail{14,1},Element_Fail{15,1},'ma w ximum*);</pre>
633	<pre>Element_Fail{16,1}=fuzzylogmath(time,alpha,Element_Fail{16,1},Element_Fail{17,1},'ma x ximum');</pre>
634	<pre>Element_Fail(19,1}=fuzzylogmath(time,alpha,Element_Fail(19,1),Element_Fail(20,1),'ma # ximum*);</pre>
635	<pre>Element_Fail{19,1}=fuzzylogmath(time,alpha,Element_Fail{19,1},Element_Fail{21,1},'ma w ximum');</pre>
63.6	<pre>Element_Fail{19,1}=fuzzylogmath(time,alpha,Element_Fail{19,1},Element_Fail{22,1},'ma < ximum');</pre>
637	<pre>Element_Fail{23,1}=fuzzylogmath(time,alpha,Element_Fail{23,1},Element_Fail{24,1},'ma < ximum');</pre>
638	<pre>Element_Fail{27,1}~fuzzylogmath(time,alpha,Element_Fail{27,1},Element_Fail{29,1},'ma w ximum');</pre>
639	Element_Fail{27,1}=fuzzylogmath(time,alpha,Element_Fail{27,1},Element_Fail{29,1},'ma # ximum');
64.0	Element_Fail{27,1}=fuzsylogmath(time,alpha,Element_Fail{27,1},Element_Fail{30,1},'ma w
641	Element_Fail{31,1}=fuzsylogmath(time,alpha,Element_Fail{31,1},Element_Fail{32,1},'ma w
642	Element_Fail{35,1}=fuzsylogmath(time,alpha,Element_Fail{25,1},Element_Fail{36,1},'ma < ximum');
643	<pre>Element_Fail{35,1}=fuzzylogmath(time,alpha,Element_Fail{35,1},Element_Fail{37,1},'ma v ximum');</pre>
644	<pre>Element_Fail{35,1}=fuszylogmath(time,alpha,Element_Fail{35,1},Element_Fail{38,1},'ma w ximum');</pre>
645	
646	\$4. Plocculation Sub-System
647	% 4 Paralel elements
64.8	8 42-43-44-45
649	<pre>Element_Fail(42,1)=fuzzylogmath(time,alpha,Element_Fail(42,1),Element_Fail(43,1), 'ma x ximum');</pre>
650	<pre>Element_Fail{42,1}=fussylogmath(time,alpha,Element_Fail{42,1},Element_Fail{44,1},'ma x ximum');</pre>
651	Element_Fail{42,1}=fuzzylogmath(time,alpha,Element_Fail{42,1},Element_Fail{45,1},'ma ×

Dec	ember 2, 2004 4:54:54 PM
32	ximum*);
652	
653	₹5. Sedimentation Sub-System
654	% 2 Paralel groups of elements
655	\$ 46-47
656	\$ 49-50
657	<pre>Element_Fail{46,1}=fuzzylogmath(time,alpha,Element_Fail{46,1},Element_Fail{47,1},'ma < ximum');</pre>
658	<pre>Element_Fail{49,1}=fuzzylogmath(time,alpha,Element_Fail{49,1},Element_Fail{50,1},'ma < ximum*);</pre>
659	
660	 Piltering Sub-System
661	<pre>% 4 Paralel elements</pre>
662	8 52-53-54-55
663	<pre>Element_Fail(52,1}=fuzzylogmath(time,alpha,Element_Fail{52,1},Element_Fail{52,1},'ma < ximum');</pre>
664	<pre>Element_Fail{52,1}=fuzzylogmath(time,alpha,Element_Fail{52,1},Element_Fail{54,1},'ma v ximum');</pre>
665	Element_Fail{52,1}=fussylogmath(time,alphs,Element_Fail{52,1},Element_Fail{55,1},'ma < ximum');
666	
667	%7. High Lifting Sub-System
668	% 4 Paralel elements
669	8 60-62-62-63
670	<pre>Element_Fail{60,1}=fuzzylogmath(time,alpha,Element_Fail{60,1},Element_Fail{61,1},'ma </pre> <pre>ximum*);</pre>
671	Element_Fail{60,1}=fuzzylogmath(time,alphs,Element_Fail{60,1},Element_Fail{62,1},'ma < ximum');
672	<pre>Element_Fail{60,1}-fuzzylogmath(time,alpha,Element_Fail{60,1},Element_Fail{63,1},'ma </pre> ximum');
673	
674	 Conveyance Sub-System
675	% there is no parallel elements
676	
677	%9. Storage Sub-System
678	\$ 65-66
67.9	<pre>Element_Fail(65,1}=fussylogmath(time,alpha,Element_Fail(65,1),Element_Fail(66,1),'ma </pre> ximum');
680	***************
681	\$*************************************

```
C:\Documents and Settings\Ibrahim\Desktop\Daily w...\rmAE.m
                                                                                Page 20
 December 2, 2004
                                                                            4:54:54 PM
682
683
      [Row_time,Col_time]=size(time);
684
685 * Rendering all Slement-Pailure MPs a full length MP on time domain.
686 for e1=1:No of Elements
687
          Element Fail{e1,1}=fuzzylogmath(time,alpha,Element Fail{e1,1},Element Fail{e1,1} #
      , 'maximum');
688
      end
689
690 % Then.....getting the start and the end of the support of each Element-Failure MF
691 # 1. Start of Support
692 for s2=1:No of Elements
693
          for f=1: (Col time-1)
694
              if (Element_Fail{s2,1}(f,2)==0) & (Element_Fail{s2,1}((f+1),2)>0)
695
                  System(s2).Element Failure Support(1,1)=Element Fail(s2,1)(f,1))
696
              end
697
          end
698
      ond
699
      $ 2. End of Support
700
      for s3=1:No of Elements
701
          for f=1:(Col time-1)
702
              if (Element_Fail{s3,1}(f,2)>0) & (Element_Fail{s3,1}((f+1),2)==0)
703
                  System(s3).Element_Failure_Support(1,2)=Element_Fail(s3,1)((f+1),1);
704
              end
705
          and
706
     end
707
      % 3. Calculation of support length of all MPs
708
      for s4-1:No of Elements
709
          System(s4).Element Failure Support Length(1,1)=(System(s4).Element Failure Support
      rt(1,2)-System(s4).Element Failure Support(1,1));
710
      end
711
712
     * Then.....getting the start and the end of the modal values of each Element-Failure «
       MP
713 # 1. Start of modal
714
     for m2=1:No of Elements
715
          for ff=1: (Col time-1)
716
              if (Element_Fail[m2,1](ff,2)<max(Element_Fail{n2,1](:,2))) & (Element_Fail{m *
      2,1]((ff+1),2)==max(Element_Fail[m2,1](:,2)))
717
                  System(m2).Element Failure Modal(1,1)=Element Fail{m2,1}(ff,1)/
718
              end
719
          end
```

Dec	ember 2, 2004 4:54:54 PM
720	end
721	\$ 2. End of Support
722	for m3-1.No of Elements
723	for ff=1:(Col time-1)
724	if (Element Fail(m3,1)(ff,2)==max(Element Fail(m3,1)(:,2))) & (Element Fail()
	m3.1)((ff+1).2) <max(element fail(m3.1)(:.2)))<="" td=""></max(element>
725	System(m3), Element Failure Modal(1,2)=Element Fail(m3,1)((ff+1),1);
726	end
727	end
728	end
729	# 3. Calculation of support length of all MPs
730	for m4-1:No of Elements
731	System(m4) Element Failure Modal Length(1,1)=(System(m4) Element Failure Modal(1)
	(2)-System(m4).Element Failure Modal(1,1));
732	end
733	
734	
735	% Build the sequence of Augmentation Sub-System step wise
736	* Connection between sub system is considered serial connection
737	System Fail Information=[System(1).Element global No(1,1) System(1).Element Failure
	Support Length(1,1) System(1).Element Failure Modal Length(1,1)
738	System (2).Element global_No(1,1) System(2).Element_Failure_S .
	upport_Length(1,1) System(2).Element_Failure_Modal_Length(1,1)
73.9	System(3).Element_global_No(1,1) System(3).Element_Failure_S
	upport_Length(1,1) System(3).Element_Failure_Modal_Length(1,1);
740	System(5).Element_global_No(1,1) System(5).Element_Failure_S
	upport_Length(1,1) System(5).Element_Pailure_Modal_Length(1,1):
741	System(7).Element_global_No(1,1) System(7).Element_Failure_S +
	upport_Length(1,1) System(7).Element_Failure_Modal_Length(1,1)
742	System(9).Element_global_No(1,1) System(9).Element_Failure_S
	upport_Length(1,1) System(9).Element_Failure_Modal_Length(1,1)
743	System(13).Element_global_No(1,1) System(13).Element_Failure w
	_Support_Length(1,1) System(13).Element_Failure_Modal_Length(1,1);
744	System (14).Element_global_No(1,1) System (14).Element_Failure w
	_Support_Length(1,1) System(14).Element_Failure_Modal_Length(1,1);
745	System(16).Element_global_No(1,1) System(16).Element_Failure (
	_Support_Length(1,1) System(16) .Element_Failure_Modal_Length(1,1);
746	System(18).Element_global_No(1,1) System(18).Element_Failure
	_Support_Length(1,1) System(18).Element_Failure_Modal_Length(1,1);
747	System(19).Element_global_No(1,1) System(19).Element_Failure +
	_Support_Length(1,1) System(19).Element_Failure_Modal_Length(1,1);
210	System (23) Element global No(1 1) System (23) Element Failure y

Dec	ember 2, 2004	cings/ibranim/besktop/bally w/rmAE.m Page 22 4154:54 PM
352	_Support_Length(1,1)	System(23).Element_Failure_Modal_Length(1,1);
749		System(25).Element_global_No(1,1) System(25).Element_Failure •
	_Support_Length(1,1)	System(25).Element_Failure_Modal_Length(1,1),
750		System (26) .Element_global_No(1,1) System (26) .Element_Failure .
	_Support_Length(1,1)	System(26).Element_Failure_Modal_Length(1,1);
751		System(27).Element_global_No(1,1) System(27).Element_Failure •
	_Support_Length(1,1)	System(27).Element_Failure_Modal_Length(1,1);
752		System(31).Element_global_No(1,1) System(31).Element_Failure .
	_Support_Length(1,1)	System(31).Element_Failure_Modal_Length(1,1);
753		System(33).Element_global_No(1,1) System(33).Element_Failure •
	_Support_Length(1,1)	System(33).Element_Failure_Modal_Length(1,1);
754		System(34).Element_global_No(1,1) System(34).Element_Failure •
	_Support_Length(1,1)	System(34).Element_Failure_Modal_Length(1,1);
755		System (35).Element_global_No(1,1) System (35).Element_Failure .
	_Support_Length(1,1)	System(35).Element_Failure_Modal_Length(1,1);
756		<pre>System(42).Element_global_No(1,1) System(42).Element_Failure •</pre>
	_Support_Length(1,1)	System(42).Element_Failure_Modal_Length(1,1);
757		System(46).Element_global_No(1,1) System(46).Element_Failure •
	_Support_Length(1,1)	System(46).Element_Failure_Modal_Length(1,1)
758		System(48).Element_global_No(1,1) System(48).Element_Failure •
	_Support_Length(1,1)	System(48).Element_Failure_Modal_Length(1,1);
759		System(50).Element_global_No(1,1) System(50).Element_Failure
	_Support_Length(1,1)	System(50).Element_Failure_Modal_Length(1,1);
760		System(51).Element_global_No(1,1) System(51).Element_Failure •
	_Support_Length(1,1)	System(51).Element_Failure_Modal_Length(1,1)
761		System(52).Element_global_No(1,1) System(52).Element_Failure .
	_Support_Length(1,1)	System(52).Element_Failure_Modal_Length(1,1);
762		System (56).Element_global_No(1,1) System (56).Element_Failure •
	_Support_Length(1,1)	System(56).Element_Failure_Modal_Length(1,1):
763		System(57).Element_global_No(1,1) System(57).Element_Failure •
	_Support_Length(1,1)	System(57).Element_Failure_Modal_Length(1,1);
764		System(63).Element_global_No(1,1) System(63).Element_Failure .
	_Support_Length(1,1)	System(63).Element_Failure_Modal_Length(1,1);
765		System (59).Element_global_No(1,1) System (59).Element_Failure •
	_Support_Length(1,1)	System(59).Element_Failure_Modal_Length(1,1);
766		System(60).Element_global_No(1,1) System(60).Element_Failure •
	_Support_Length(1,1)	System(60).Element_Failure_Modal_Length(1,1);
767	1953 52	System(65).Element_global_No(1,1) System(65).Element_Failure •
	Support Length(1,1)	System(65) Element Failure Modal Length(1,1)]:

Dec	ember 2, 2004 4:54:54 Pi				
769					
769	& Determine the controling WF. (first MF with max support, then MF with max				
770	* modal)				
771	· Included a				
772	[Row Fail,Col Fail]=size(System Fail Information);				
773					
774	max support=max(System Fail Information(;,2));				
775	max support elements-find(System Fail Information(1,2)max support);				
776	[Row max support elements.Col max support elements]=size(max support elements);				
777					
778	if (Row max support elements>1)				
77.9	max modal-max(System Fail Information(:,3));				
780	<pre>max_modal_elements=find(System_Fail_Information(:,3)==max_modal);</pre>				
781	[Row max modal elements, Col max modal elements]=size(max modal elements);				
78.2	if (Row max modal elements>1)				
783	System_Fail=[System_Fail_Information(max_modal_elements(1,1),1) System_Fail_				
	Information(max_modal_elements(1,1),2) System_Fail_Information(max_modal_elements(1,				
	1),3)],				
784	end				
785	System_Fail={System_Fail_Information((max_modal_elements(1,1)==max_modal),1) Sys				
	tem_Fail_Information((max_modal_elements(1,1) == max_modal),2) System_Fail_Information				
	((max_modal_elements(1,1) == max_modal),3));				
786	end				
787					
788	System_Fail= System_Fail_Information(max_support_elements(1,1),1) System_Fail_Inform				
	ation(max_support_elements(1,1),2) System_Fail_Information(max_support_elements(1,1)				
	,3)];;				
789					
790	***************************************				
791	***Step TEN-FOURuse the MF of the controling Element				
792	System_Fail_MF=Element_Fail{System_Fail(1,1),1};				
793					
794	<pre>[row_t,col_t]=size(System_Fail_MF);</pre>				
795					
796	Resilience_Index=0;				
797	nominator=0;				
798	denominator=0;				
799	for t=1:(row_t-1)				
800	nominator=nominator+(0.5*(System_Pail_MF(t+1,1)+System_Pail_MF(t,1)));				
801	<pre>denominator=denominator+(System_Fail_MF(t,2)*(0.5*(System_Fail_MF(t+1,1)+System_</pre>				
	Fail_MF(t,1))));				
803	end				

803 Resilience_Index-nominator/denominator

II-C CUSTOM MATLAB FUZZY SCRIPTS FUZZY TRIANGULAR MEMEBRSHIP FUNCTION FUZZY TRAPIZOIDAL MEMEBRSHIP FUNCTION

3. FUZZY ARTITHEMTATIC OPERATIONS

4. FUZZY LOGIC OPERATIONS

```
C:\Documents and Settings\Ibrahim\Desktop\D...\triangular.m
                                                                              Page 1
                                                                         5:00:56 PM
December 2, 2004
 1 function A=triangular(alpha,a,b,c)
 2
    %TRIANGULAR Fuzzy Number
 3
    # Last Modified; Sunday, July 25, 2004 @6:15pm
 4
 5 if (b<a),
 6
       disp('Not valid input - b must be >= a');
 7
        break;
 8 elseif (c<b),
 9
       disp("Not valid input - c must be >= b");
10
       break;
11 end
12
13
14
    [Row_al,Col_al]=size(alpha);
15
16 No_alpha_step=Col_al-1;
17
    alpha_step=max(alpha)/No_alpha_step;
18
19
         for 1-1:((2*No alpha step)+1)
20
                if (i<=No alpha step)
21
                    MFN(1,1)=a+((1-1)*alpha step*(b-a));
22
                    MFN(1,2)=min(alpha)+((i-1)*alpha_step);
23
                else
24
                    MFN(1,1)=b+((1-1-No_alpha_step)*alpha_step*(c-b));
25
                    MFN(1,2)=max(alpha)-((1-1-No alpha step)*alpha step);
26
                and
27
         and
28 A= [MFN (:,1) MFN (:,2)];
```

```
C:\Documents and Settings\Ibrahim\Desktop\...\trapizoidal.m
                                                                             Page 1
December 2, 2004
                                                                         5:00:03 PM
 1 function A=trapizoidal (alpha,a,b,c,d)
2 %TRAPIZOIDAL Fuzzy Number
 3 & Last Modified: Sunday, July 25, 2004 @7:00pm
 4
 5 if (b<a),
 6
       disp('Not valid input - b must be >= a');
 7
       break:
 8 elseif (c=a),
 9
       disp("Not valid input - c must be >= a');
10
       break;
11 elseif (d<a),
       disp("Not valid input - d must be >= a");
12
13
       break;
14 elseif (c<b),
15
       disp('Not valid input - c must be >= b') r
16
       break;
17 elseif (d.b),
18
       disp("Not valid input - d must be >= b');
19
        break;
20
    elseif (d<c),
21
        disp('Not valid input - d must be >= p');
22
       break;
23 end
24
25
26
    [Row al, Col al]=size(alpha);
27
28 No alpha step=Col al-1;
29 alpha_step=max(alpha)/No_alpha_step;
30
31
        for 1=1:((2*No_alpha_step)+2)
32
                if (i<=No_alpha_step)
33
                    MFN(i,1)=a+((i-1)*alpha_step*(b-a));
34
                    MFN(i,2)=min(alpha)+((i-1)*alpha step):
35
                end
36
37
                if (i==(No_alpha_step+1))
38
                    MFN(1,1)=b;
39
                    MFN(1,2)=1;
40
                end
41
42
                if (i==(No_alpha_step+2))
```

December 2, 2004					PN
43			MPN(1,1)=c;		
44			MFN(1,2)=1;		
45		end			
46					
47		if	(i>(No_alpha_step+2))		
48			<pre>MPN(i,1)=c+((i-2-No_alpha_step)*alpha_step*(d-c));</pre>		
49			MFN(1,2)=max(alpha)-((1+2-No_alpha_step)*alpha_step);		
50		end			
51	end				
52	A= [MFN (: ; 1)	MFN	(;,2)];		

C:\Documents and Settings\Ibrahim\Desktop\...\trapizoidal.m Page 2 M

```
C:\Documents and Settings\Ibrahim\Desktop\Da...\fuzzymath.m
                                                                               Page 1
December 2, 2004
                                                                          5:00:25 PM
 1
     function result = fuzzymath(alpha,A,B,operator)
 2
     & Last Modified: Monday, July 26, 2004 @9:00am
 3
 4
 5 [Row_al,Col_al]=size(alpha); * basic information about membership values (alpha)
 6 No alpha step=Col al-1;
 7
    alpha_step=max(alpha)/No_alpha_step;
 B
 9 & Basic information about the two membership functions
10 [Row_A,Col_A]=size(A);
11 [Row_B,Col_B]=size(B)/
12
13 & a loop to make one matrix with left and right value to perform the interval calcul .
     ations (this is done for the
14
    $% two membership functions)
15
        * the LEFT values
16
    for i=1:Col al
         AA(1,2)-A(1,1);
17
18
         BB(1,2)-B(1,1),
19
     end
20
21
         % the RIGHT values
22 counter=0;
23
    for j=0:No_alpha_step
24
         counter=counter+1;
25
         AA(counter,3)=A((Row A-j),1);
26
         BB(counter,3)=B((Row_A-j),1);
27
     end
28
         * corresponding alpha values
29
    for k=1:Col al
30
         AA(k,1)=A(k,2);
31
         BB(k,1)=B(k,2);
32
    end
33
34 § Interval arithmatic
35 for L=1:Col al
36
         if stromp(operator, 'sum'),
37
                result_interval(L,1)=A(L,2);
38
                result interval(L, 2) =AA(L, 2) +BB(L, 2);
39
                result interval(L,3)=AA(L,3)+BB(L,3);
40
41
         elseif stromp(operator, 'sub'),
```

```
C:\Documents and Settings\Ibrahim\Desktop\Da...\fuzzymath.m
                                                                                Page 2
December 2, 2004
                                                                           5:00:25 PM
42
                 result_interval=[(AA(L,1)-BB(L,2)) (AA(L,2)-BB(L,1))];
43
                 result_interval(L,1)=A(L,2);
44
                 result_interval(L,2)=AA(L,2)-BB(L,3) /
45
                 result_interval(L,3)=AA(L,3)-BB(L,2);
46
47
         end
48
     end
49
50
51 * Rebuilding the resultant Membership function in the same way as in the
    % input membership functions
52
53
         % the Triangular case
54 if (A(Col_al,2)==1) & (A((Col_al+1),2)==1)
55
         for y1=1:Col_al
56
             result(y1,2)=A(y1,2);
57
             result(y1,1)=result_interval(y1,2);
58
         ond
59
         for y2-1:No alpha step
60
             result((Col al+y2),2)=A((Col al-y2),2);
61
             result((Col_al+y2),1)=result_interval((Col_al-y2),3);
6.2
         ond
63
         % the Trapisoidal case
64
     elseif (A(Col_al,2)==1) & (A((Col_al+1),2)==1)
65
         for z1=1:Col al
66
             result(21,2)=A(21,2);
67
             result(z1,1)=result interval(z1,2);
68
         and
69
         for z2=0:No alpha step
70
             result((Col_al+z2+1),2)=A((Col_al-z2),2);
71
             result((Col_al+z2+1),1)=result_interval((Col_al-z2),3);
72
         and
73
     end
74
75
76
```

```
C:\Documents and Settings\Ibrahim\Desktop...\fuzzylogmath.m
                                                                          Page 1
December 2, 2004
                                                                      5:00:44 PM
     function C = fuzzylogmath(Domain, alpha, A, B, operator)
 1
 2
    & Last Modified: Friday, August 19, 2004 @8:00pm
 3
 4
 5 [Row_al,Col_al]=size(alpha); * basic information about membership values (alpha)
 6 No alpha steps=Col al-1;
 7
    alpha_step=max(alpha)/No_alpha_steps;
 A.
 9
10 [Row_Domain,Col_Domain]=size(Domain); * basic information about Domain-values (Domai &
    n-axis)
11 No_Domain_steps=Col_Domain-1;
12 Domain step=max(Domain)/No Domain steps;
13
14
15 ###-----Step-ONE----- (Modifying NFs)
16
    $58 ·····
17
        % Add end values for A and B to cover the whole range of Domain-values....to avo 
     id giving negative values in case
18
        $% of not defining the end Domain-values of A and B.
19
20
        if min(Domain) < A(1,1)
21
            AA(1,1)=min(Domain);
22
            AA(1,2)=0;
22
            A=[AA(1,1) AA(1,2);A];
24
        end
25
    [Row A, Col A]=size(A); * Basic information about the MP of A, it is here to take in #
     to account any change
26
                           # due to the change in dimensions after the first condition
27
28
        if max(Domain) >A(Row A, 1)
29
            AA((Row_A+1), 1) = max(Domain))
30
            AA((Row A+1),2)=0;
31
            A=[A; AA((Row A+1), 1) AA((Row A+1), 2)];
32
        and
33
34
        if min(Domain)<B(1,1)
35
            BB(1,1)=min(Domain);
36
            BB(1,2)=0;
37
            B=[BB(1,1) BB(1,2);B];
38
        end
39
    [Row_B,Col_B]=size(B): ∦ Basic information about the MF of B, it is here to take in ≤
```

```
C:\Documents and Settings\Ibrahim\Desktop...\fuzzylogmath.m
                                                      Page 2
December 2, 2004
                                                   5:00:44 PM
   to account any change
40
                   * due to the change in dimensions after the first condition
41
42
      if max(Domain)>B(Row B,1)
43
        BB((Row B+1),1)=max(Domain);
44
        BB((Row B+1),2)=0;
45
        B=[B;BB((Row_B+1),1) BB((Row_B+1),2)];
46
      end
47 **-----
48 88-----
49
50 $$$------Step---TWO-------(Interpolate for Domain steps)------
51 ***-----
52 A=interpl(A(:,1),A(:,2),Domain);
53 A=[Domain:A];
54 A=A';
55
56 B=interpl(B(:,1),B(:,2),Domain);
57
   B= [Domain (B) ;
58 B+B';
59 **-----
60 88-----
61
62
   ###-----Step---THREE----- (Identify the operator and perform operation)
63
   64
65
   if stromp(operator, 'minimum'),
66
      C-min(A(:,2),B(:,2));
67
   end
68
69
   if stromp(operator, 'maximum'),
70
      C=max(A(:,2),B(:,2));
71
   end
72
73 C=[(Domain') C];
74
75
   [Row C,Col C]=size(C);  # Basic information about the MF of C
76 for w=1:Row_C
77
      if C(w,2)<0
78
        C(W, 2) = 0
79
      end
80
   end
```