Contents

1 Emergency Management System LiveDesign

1.1 LiveDesign

LiveDesign is a computing intensive hardware-software integrated environment. The hardware consists of computer and external measuring devices, *e.g.* sensors, monitors, microphones. The data stream may also come, for example, from computer communication traffic, the energy grid, stock market data, medical record and image transaction, and entertainment videos and three-dimensional television broadcast.

In order to save the maximum number of human lives, LiveDesign aids crisis management authorities to mitigate disasters in real-time. Based on the past experiences, LiveDesign is capable of predicting future disasters and can suggest the precautionary and rescue operations that can be efficiently initiated and carried out before, during and after impending disasters that could cripple the socio-economic civic live. As a natural, industrial, manmade disaster or any combination of those unfolds LiveDesign broadcasts and documents an up-to-date safety and security characterization of physical infrastructure systems geared with compatible monitoring, sensing and surveillance devices.

In health care industry, for example, the motion of a patient or a senior citizen, for example, can be monitored with infrared sensors and analyzed for a fall that is about to happen. An instantaneous messaging to the emergency medical service provider can bring quick help for a victim of a stroke or heart attack, for example. Similarly, some not-so-obvious man-made disasters, *e.g.* cable leaks from the US State Department, may be prevented by monitoring unusual traffic patterns in document downloading streams. Similarly, in the field of economic analysis unusual financial collapses, that happen once say in forty or fifty years, can be apprehended by detecting the extreme value statistical signatures in financial transactions.

The same software with the appropriate LiveDesignDataBase is capable of compressing two- and three-dimensional video streams. For example medical images are usually very large. Dynamic transmission across the country encounters and accumulates noise and errors. The anatomical structure in those images can be recognized using medical knowledge and geometrical feature identifiers. The shape and size descriptors are transmitted instead of bits on the x-ray or CAT Scan, for example, to gain speed and assure accuracy.

1.1.1 LiveDesignKnowledgeBase

The LiveDesignDataBase combines all quantitative and qualitative data, *e.g.* engineering and policy information. In this example, engineering specifications consist of numbers, technical descriptions, charts and diagrams. Policy statements and bodies of laws that are written in natural languages. LiveDesignDataBase organizes and combines all such pieces of information in a situation dependent (*live*) LiveDesignKnowledgeBase. The LiveDesignComputationalEngine also contains all search, operational and function–transformation computer program modules (*design* tools) for real-time information processing. In this document all such computing is performed using the computer algebra environment (a commercial computing program entitled *Mathematica*).

The name LiveDesign is the embodiment of the *live* aspect of emergency management *design* that actively optimizes mitigation in real-time.

1.1.2 Some Parts of a generic LiveDesignComputationalEngine

1.1.2.1 Deterministic Computation

Standard libraries of numerical routines to solve algebraic and differential equations, which are available universally at all scientific computer centers, can be used. These libraries of computer programs provide support to build applications such as game theory, network theory, combinatorics, optimization and graphics. The LiveDesign calculations presented here are enhancements described in detail in §4 and §5.

1.1.2.2 Probabilistic Numerical and Symbolic Computation

The two parts are Probabilistic Numerical Computation and Probabilistic Symbolic Computation. The later uses computer algebra for exact representation without approximation of mathematical expressions for probability distributions and their functions.

1.1.2.3 Fuzzy Logic, Game Theory and Bayesian Updating

Symbolic algebraic computation and numerical arithmetical calculations are two parts used in Fuzzy Logic computations presented here. Fuzzy Logic Numerical Computation, and Fuzzy Logic Symbolic Computation use computer algebra to address higher order, such as type-2 fuzziness, in Game Theory and Bayesian Updating modules.

1.2 Schematic of Conceptual Modules and Data Flow

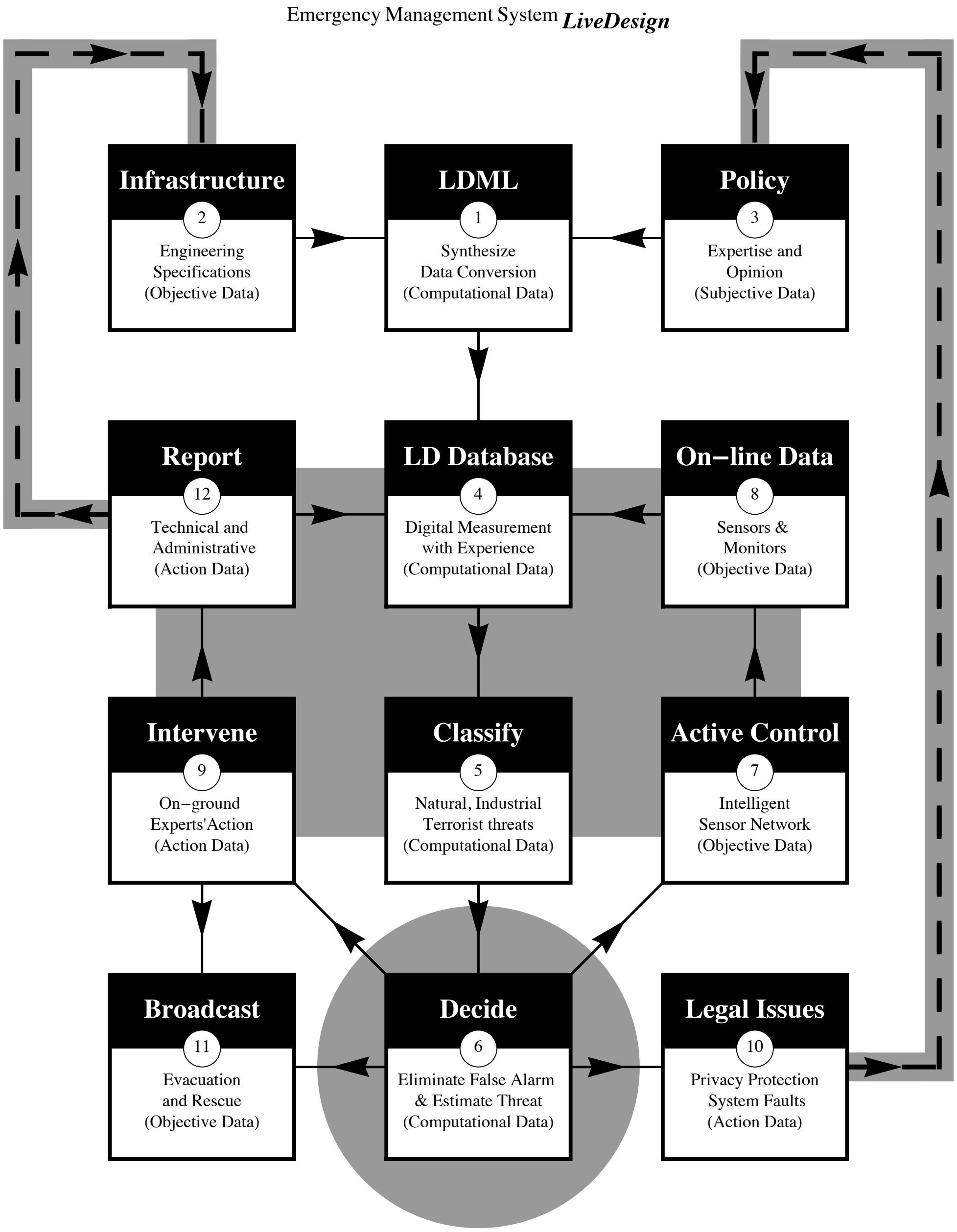


Figure 1.1: LiveDesign Emergency Management System

There are the three following aspects of integrating objective, subjective, computational and action data, in LiveDesign schematically shown in Figure 1.1, to aid emergency management authorities:

(i) preparation for mitigation: modules labeled 1, 2 and 3;

(ii) human intervention: modules 9 and 10;

(iii) automated computing cycle: modules 4, 5, 6, 7, 8, 11 and 12.

Brief summaries of (i), (ii) and (iii) are furnished as follows. Therein, the module group (iii) aids (ii) during an emergency described in(i).

1.2.1 Preparation for mitigation: the pre-requisite

Each situation requires an objective, precise, exact and crisp non-ambiguous technical description of the infrastructure in module 2, and the governing laws, which may need interpretation, experts' subjective opinions, which are not universally agreed upon, and the safety and security regulations, which can vary from place to place, of such infrastructure in module 3. A dictionary type table look up and data conversion and data interchange module is 1 that integrates the information in modules 2 and 3. To expedite data input in actual use and to configure the system initially the sensor network, which is a computing system in tandem with sensor and monitors, the entire system described in Figure 1.1 may be used.

1.2.1.1 Implementation of available computer languages

If a dedicated front-end interface is used it may be customized to suit the LiveDesign computing facilities described in Figure 1.1. For example, the web browser, which is a popular interface, may be used when the language of choice could be Java or Java++. Similarly in many other interfaces, for example,, C or C++, may be selected.

1.2.1.2 Implementation of LiveDesign Markup Language

This is a special purpose extension of the standard XML. Additional computer program elements are added to use concepts related to mitigating emergencies in the particular infrastructure. The complete LiveDesign Markup Language in Figure 1.1 resides in module 1 and is distributed from there.

1.2.2 Human intervention

Experienced emergency managers must be able to undertake on the spot action via technical command control module 9 and the regulation scrutinizing module 10. Touch-screen and voice interface technologies adapt to computing systems that executes the emergency management computing cycles by looping through the modules, for example, according to those paths shown in Figure 1.1. Hence, this aspect does not constitute the subject matter here and thus this aspect is not elaborated in this application.

1.2.2.1 Input from the public

Broadcast directives can be corrected when the citizens using the facilities, such as the evacuees, can communicate with emergency management staff on the ground using custom mobile device programs (apps) in addition to directly approaching the ground crew.

1.2.2.2 Input from remote observers

Man policy makers, emergency specialists will receive the broadcast messages at remote sites along with texts, graphics, sound describing the emergency event. If they catch any deficiency in the current mitigation then can contact module 9 to rectify the error.

1.2.3 Automated computation cycle

Data collection via sensors and monitors in module 8, may be instantaneous but there is some real time needed, t1 seconds, to write it in the LiveDesignDataBase. Stepping through different algorithms in modules 4, 5, 6 and 7 will need t2 seconds. With an overhead of t3, for example, in parallel processing, to run the system each computing cycle will be of totalTime=t1+t2+t3 seconds in duration. The conventional technical term "real time," will refer to totalTime, not 0 seconds between observation and action.

$$\text{data input time in LiveDesignDataBase} : \text{t1} \tag{1.1}$$

$$\text{time to execute LiveDesign algorithms} : \text{t2} \tag{1.2}$$

$$\text{overhead time} : \text{t3} \tag{1.3}$$

$$\text{total time for one computing cycle} : \text{totalTime} = \text{t1+t2+t3} \tag{1.4}$$

1.3 Elements of the Emergency Knowledge System

1.3.1 Steps in conceptual LiveDesign network

The data flow network through 12 modules of Figure 1.1 is conceptually represented by a block diagram, which facilitates the step by step depiction, entitled the Emergency Knowledge System, which is labeled number 20, in Figure 1.2.

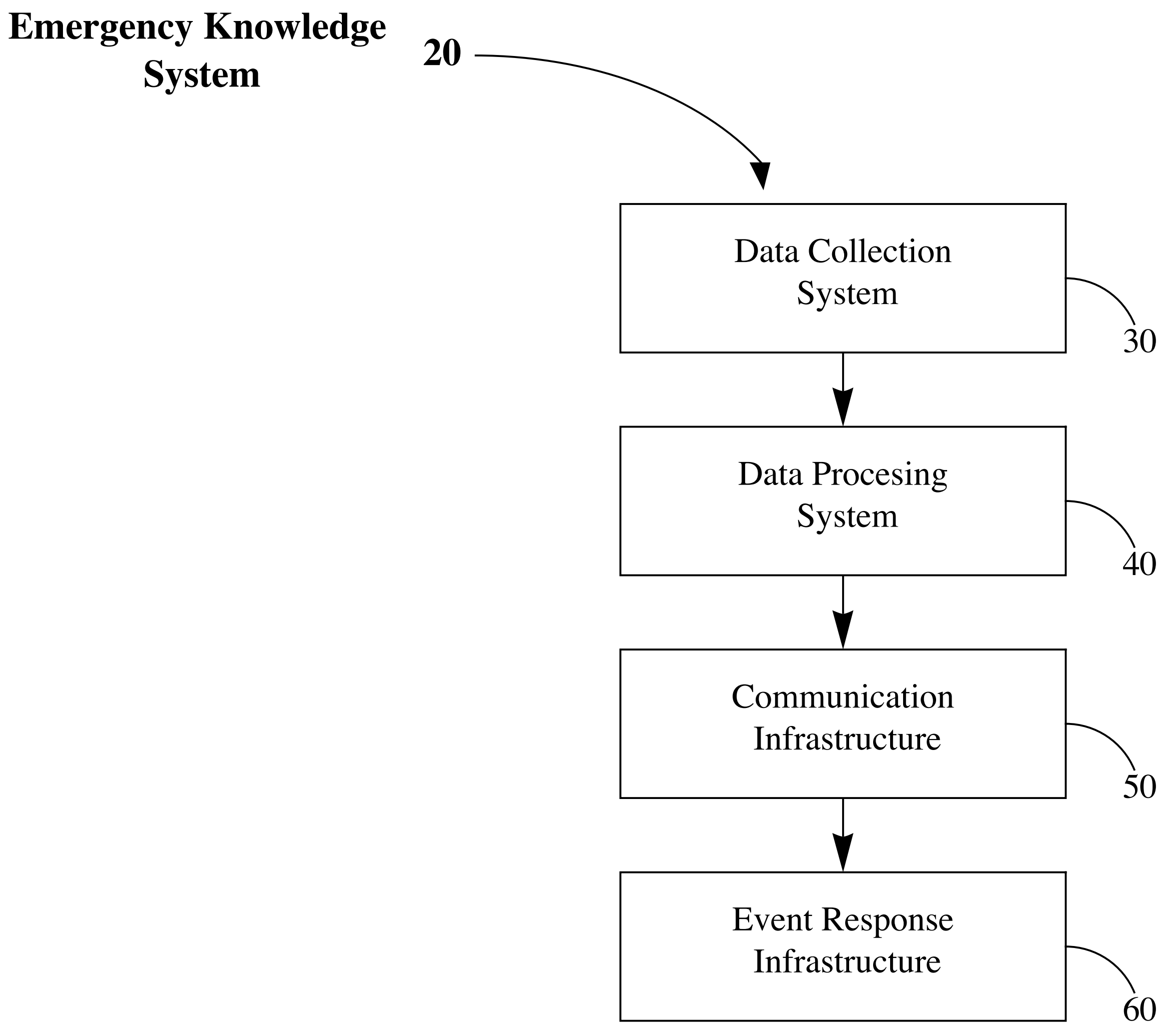


FIG. 2

Figure 1.2: Emergency Knowledge System, block number 20

The block diagram shown above contains units marked 30, 40, 50 and 60. Each unit with a unique identifier has a short description within the rectangle. The details of the four units 20, 40, 50 and 60 are further expanded in block diagrams in section 2. All short descriptions, one appears within each unit, are explained in section 3.

1.3.2 Input, Output and the Main LiveDesign Loop

It may be noted that the steps illustrated in Figure 1.3 are associated with the units described in Figure 1.2.

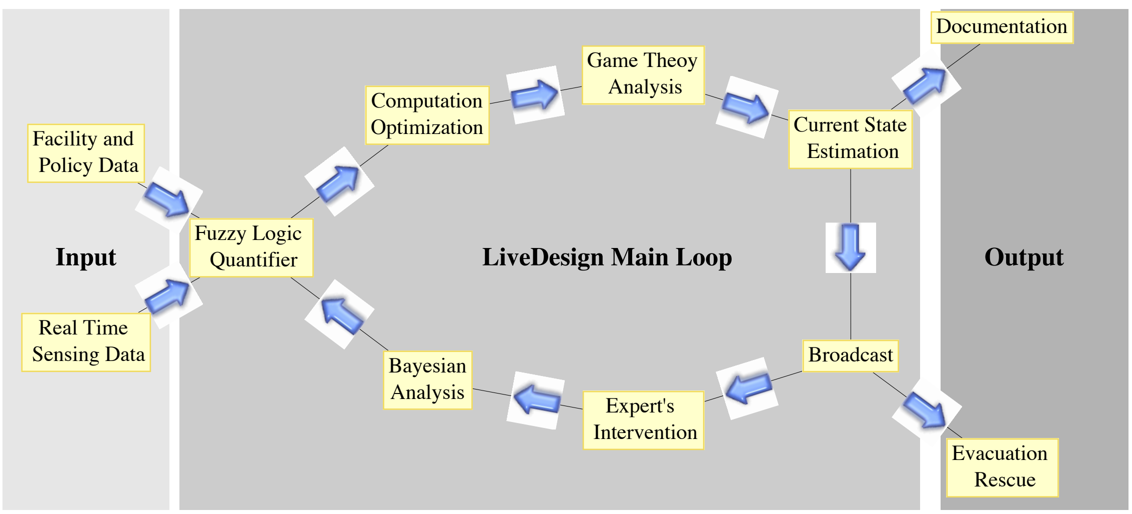


Figure 1.3: Input, Output and the Main LiveDesign Loop

The schematic in Figure 1.3 is associated with different topics, which are details are elaborated in §4, of computing science and applied mathematics. Depending upon a problem, in addition to which are addressed by the local civic authorities and those massive ones for which the state and national level of emergency management should get involved, one such topic may dominate. Some examples follow.

1.3.2.1 Stampedes in Large Gatherings

For example, the fuzzy logic unit will work more in a situation like stampede in a sports arena the intensity of danger could be more than what is observed during normal traffic accident, fire in an apartment etc. but less than the highway traffic jam after Katrina. The fuzzy logic unit in those cases will be designed to assign a fraction (membership function) that will indicate the extent the local management may be inadequate.

1.3.2.2 Mass Transit and Transportation Accidents

Another example could be an accident in a railway track where the evacuation from a platform or the region surrounding the accident site may require much more vigorous second by second updating of the emergency management strategy. In that case the Bayesian logic unit may dominate in computational tasks since this is the workhorse that updates all uncertain information in the LiveDesignDataBase.

1.3.2.3 Terrorist-made Disaster

In a terrorist attack, the Game Theory calculations should be very important so that the next move by the terrorist can be averted and can be used to capture the miscreants.

1.3.2.4 Senior Housing

Senior citizens housing are serviced by on-site health care providers. For example, a fall can be detected by a sensing system. LiveDesign may broadcast the accident to the medical unit along with the medical record. This will help determine, for example, whether the fall could have been due to a heart attack or a slip at a crumpled rug. The LiveDesignDataBase will have personal medical data which may be matched with apprehended health conditions for that particular person. Now whether to summon a paramedic or a heart specialist could be suggested by the LiveDesign computing hardware-software system. By shortening the arrival time of help lives may be saved.

1.3.2.5 Evolving Physical Description during Constructions

When a bridge, building or a ship is being constructed the local emergency management authority does not possess an up to date picture of the on-going construction. So at a disaster the final finished facility from engineering drawings do not help the first responders. Following the direct input of construction schedule and from video and other monitors the LiveDesignDataBase may construct an exact 3-d picture of the current state of the building, bridge, ship, as the case may be. These graphics and texts can be transmitted to the emergency management authority as well as to the people in danger in the premises. The secure and evacuation operation will improve.

Broadcasting engineering design blocks for 2-d and 3-d display screens is facilitated by the data compression described in sec:data.compression. For 3-d transmission, following equation (**??**) the estimated efficiency in data transmission could be:

$$\text{2-d compression} = 700, \text{from: equation } (\mathbf{??})$$
$$\text{estimated 3-d compression} = (\sqrt{700})^3 = 18520.3 \approx 18{,}500 \text{ times}$$

1.3.2.6 Cyber-secured Diagnostics for the Energy Grid

Wired with the LiveDesign hardware-software active system safe functioning of the energy distribution system may be monitored. Any malfunction or deliberate attack can be identifies in no time. Self-powered intelligent sensor network may send codes to the command-post where LiveDesignDataBase will decipher codes to text and graphics.

1.3.2.7 Text and Image Transmission to First Responders

The fast arriving 3-d images to the first responders as they travel from the command-post to the endangered site prepare them to take immediate actions upon arrival. These dedicated, highly secured, communication streams can be transmitted globally with almost no power consumption at the disaster location. It is then possible to instruct the habitants from the command post how to minimize personal injuries, for example.

1.3.2.8 Medical Image and Video Stream Transmission

Medical images have a special property that the graphics depict well defined anatomical features. An ordinary OCR *optical character recognition* program do not have the geometrical structure that adheres to the biology of the anatomical object described in the transmission data. A LiveDesignDataBase can be written for a human liver, for example, if the CAT Scan of the liver is transmitted; similarly a lung LiveDesignDataBase will facilitate transfer of lung scans. These LiveDesignFeatureExtractors help extract the feature data and transmit the following information:

$$\text{feature-1} : value_1 \tag{1.5}$$

$$\text{feature-2} : value_2 \tag{1.6}$$

$$\text{feature-3} : value_3 \ldots \tag{1.7}$$

$$\text{feature-i} : \text{a biological term for a feature} \tag{1.8}$$

$$value_i : \text{a numerical value extracted for the feature-i} \tag{1.9}$$

Instead of sending all those pixel data, say 100 feature names and their numerical values are transmitted maybe in a coded form for data security. On the receiving side the associated LiveDesignDataBase, different for the liver and lung from the above example, when executed can reconstruct the image without any loss in medical information. Specialists trained in anatomy and familiar geometry of the particular anatomical element will furnish the subjective information that are typically stored in module number 3 in Figure 1.1.

For a video transmission the same compression algorithm is applied for each frame.

1.3.2.9 Unusual traffic pattern

Automobile traffic tie ups, abnormal movement and the loss of balance of a patient just before a collapse, unauthorized downloading volumes of commercial proprietary and government sensitive data, for example,, follow very similar time signature that can be detected a very similar LiveDesign engine with different special purpose LiveDesignDataBase.

All the above examples may be executed through stages in Figure 1.3

2 Block Diagrams

2.1 Block Diagrams with Conceptual Modules

The twelve conceptual modules, shown in Figure 1.1, in the complex network of data flow, perform multiple actions, as indicated by their possible multiple appearances, in emplaning the block diagram of Figure 1.2. The four units of Figure 1.2 are explained in Figure 2.1, Figure 2.2, Figure 2.3 and Figure 2.4, as follows. The technical terms used in Figures 4, 5, 6 and 7 are defined in section 3.

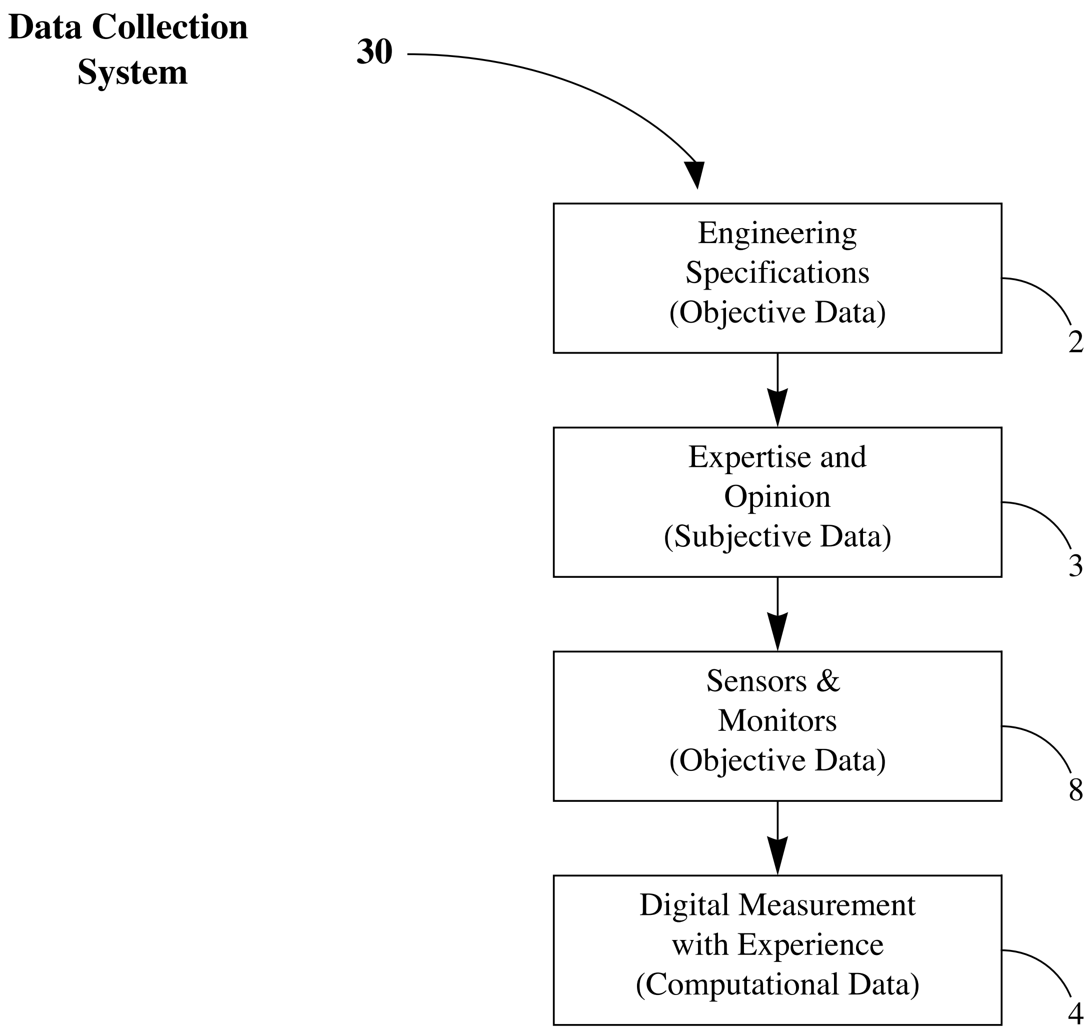


FIG. 4

Figure 2.1: Data Collection

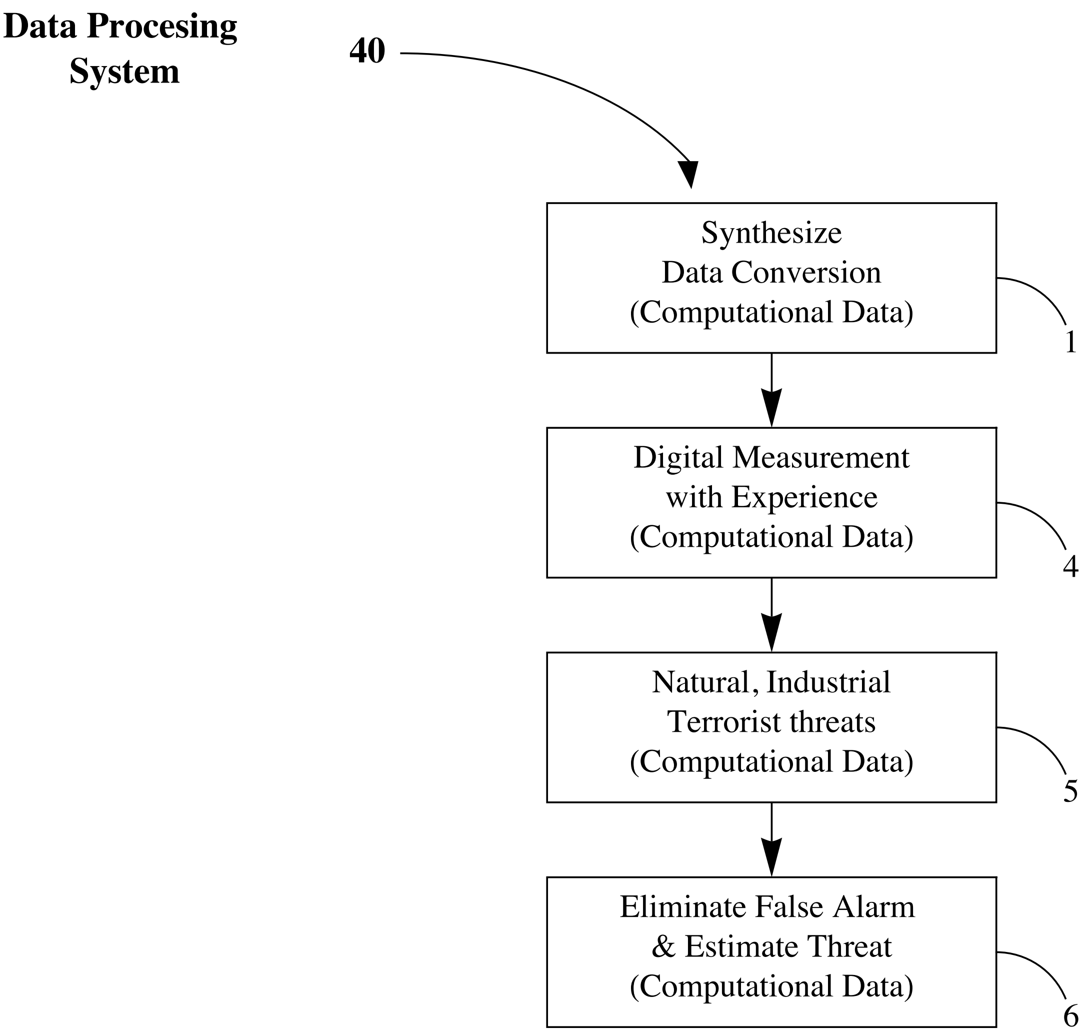


FIG. 5

Figure 2.2: Data Processing

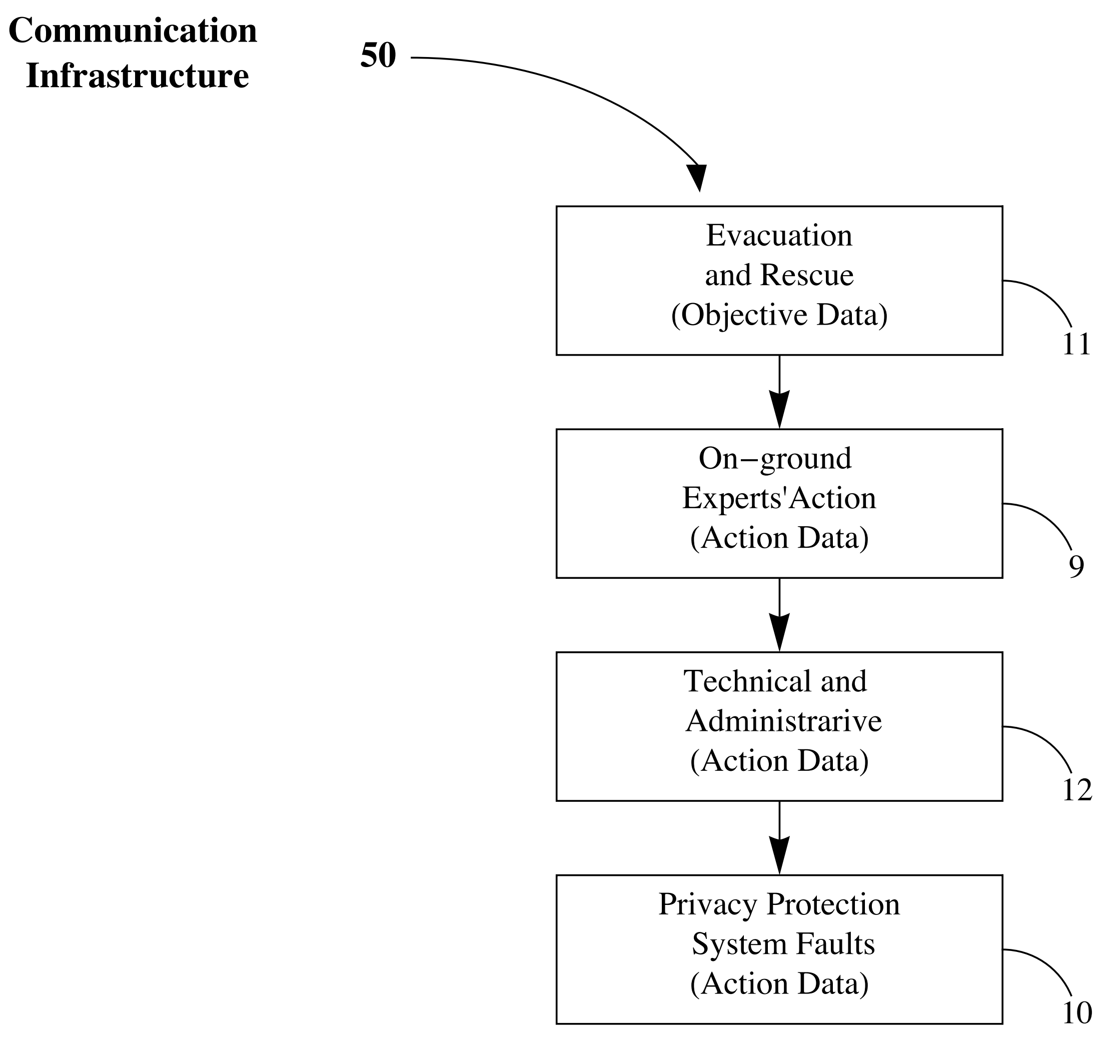


FIG. 6

Figure 2.3: Communication

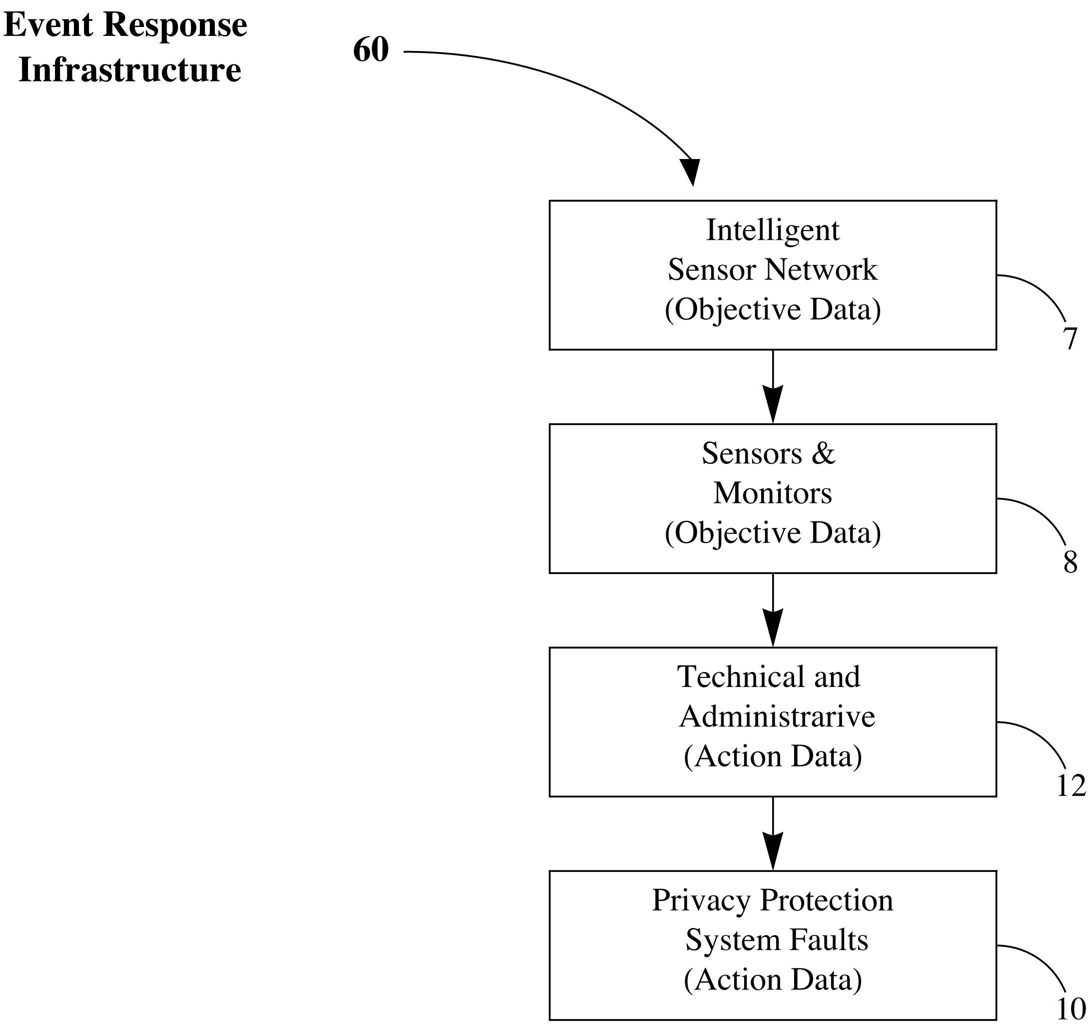


FIG. 7

Figure 2.4: Emergency Response

3 Definitions

The LiveDesign computer programs loop through the following four types of data:

(a) Objective Data

(b) Subjective Data

(c) Computational Data

(d) Action Data

There are three types of objects used in figures, they are:

(i) the twelve conceptual modules, shown in Figure 1.1

(ii) the block diagrams, shown in Figure 1.3 through Figure 2.4

(iii) the units that appear in block diagrams, shown in Figure 1.3 through Figure 2.4

The three unique features, which can be explained by the modules, blocks and units as mentioned above, are:

(A) Intelligent Sensor Network

(B) Elimination of False Alarm

(C) Early Warning System

3.1 Different Types of Data

3.1.a Objective Data

This is an assemblage of:

(I) Engineering Specifications, module number 2

(II) Digital Data stream coming out of sensors and monitors, module number 7

(III) Time dependent instructions to the sensors and monitors focusing what to look for, module number 8

(IV) Digital Data stream coming out of the broadcast mechanisms to warn the human users how to take safety steps, module number 11

Objective Data encompasses all such variables.

The textual parts are written as is in the LiveDesignDataBase. The Graphics are stores as encapsulated postscript files, `.eps`.

3.1.b Subjective Data

There is only one item:

(V) Module number 3 houses non-engineering management type of information.

Experiences of emergency management authority play a significant role in mitigating any out of the ordinary situation. On the same issue experts may have different opinion. Some experts may have more credibility on certain aspects. All these inexact characterizations are digitized using fuzzy logic constructs. Since an opinion can be other than all correct or all wrong, fuzzy logic representation of partial truth for everything, which can have two different answers, is encased within a subjective dataset.

A subjective data piece is written as a fuzzy logic element that is associated with a membership function,

3.1.c Computational Data

As the emergency related information is streaming into the LiveDesign Management system, the subjective and objective data are utilized side by side resulting in computational data that flows through the LiveDesign computational cycles. The modules that perform computational tasks are denoted as follows:

(VI) organize all objective and subjective data under one electronic digital file, module number 1

(VII) continuously update, in real-time, the current information, and write the information in LiveDesign Markup Language in LiveDesignDataBase designated by module number 4

(VIII) classify data for more effective mitigation according to natural, man-made or industrial disaster in module number 5

(IX) eliminate false alarm and program the smart sensor network to focus attention on a certain type of data, such as a chemical content in the air over the temperature distribution in module number 6

3.1.d Action Data

The outcomes of running a LiveDesign computer program is encapsulated within the Action Data. The outcomes are:

(X) to furnish optimal mitigation suggestions to emergency management authority, module number 9

(XI) to compile a report on the effectiveness of executing the LiveDesign computer program, module number 12

(XII) to detect an inconsistency in policy issues, module number 10

3.2 Conceptual Modules

3.2.1 Synthesize Data Conversion

LDML: LiveDesign Markup Language is an extension of XML[1] that is used to put all subjective and objective information within the same electronic file. The conversion of all subjective and objective data is carried out only once before the system is commissioned.

3.2.2 Engineering Specifications

Infrastructure: Engineering drawing, such as floor plans, design data, such as the load and temperature on a structure, can be exactly represented. Even when there is a tolerance of 0.1% or so, digitized versions do not cause any difference in opinion amongst experts when a certain expected number is stated to lie within an interval, which is expressed by some agreed upon upper and lower limits, that signifies the tolerance of the value.

3.2.3 Expertise and Opinion

Policy: The emergency management authorities use two different modes of mitigation for:

[1]Reference: Extensible Markup Language (XML) is a set of rules for encoding documents in machine-readable form. It is defined in the XML 1.0 Specification produced by the W3C, and several other related specifications, all gratis open standards

1. Routine Emergency Calls: managed by local authorities, with fire, police and ambulance departments
2. Extreme Emergency Situations: like Hurricane Katrina, BP Gulf Oil Spill and 9/11, for which national and international rescue and mitigation efforts are essential

3.2.4 Digital Measurement with Experience

LD Database: A unified electronic file combines all numerical data and expert opinion. The latter is transcribed into a string of numbers using fuzzy logic to translate each piece of opinion into a string of alpha-numeric text.

3.2.5 Natural, Industrial Terrorist threats

Classify: In an actual disaster experts decide the severity, say in a scale of red (most severe), amber, yellow, blue and green:

Figure 3.1: Homeland Security Classification of Threats

that red and yellow will indicate a security violation as opposed to blue and green that will signify safety violation, which can be handled by local authorities.

For the extreme cases of red and amber alert the current data stream can be classified to fit into one of the three categories of extreme statistics:

- • Natural disaster
- •• Terrorist threat
- ••• Industrial failure

Extreme value statistics[2] provides three type of distributions which are used in LiveDesign computation as follows:

- □ Type-1: Gumbel Distribution: Natural disaster
- □□ Type-2: Fréchet Distribution: Terrorist threat
- □□□ Type-3: Weibull Distribution: Industrial failure

These types of distributions are also used in generating simulation samples to test the LiveDesign computation engine

3.2.6 Eliminate False Alarm & Estimate Threat

Decide: It can be widely observed that Safety and Security are imprecise overlapping concepts. Safety and Security are natural language terms that experts frequently use within the context of emergency management. The notion of fuzzy set membership functions, which was proposed by Zadeh, provides a useful numerical tool to quantity the degree of violation of safety and or security in an emergency.

Fuzzy logic tell us how to construct these Membership Functions such as using frequencies of subjective human answers. We can poll several experts to ascertain a color coding red, yellow, blue and green to broadcast the severity of an emergency.

Mathematically the concept of probability deals with mutually exclusive non-overlapping events. In terms of theoretical mathematics logic and probability theory assume exact and well-defined predicates and classical set-theoretic semantics and the notion is formalized with Kolmogorov's axioms. Probabilities are akin to frequencies of repeating events. Some severe emergencies do not occur frequently, so subjective judgement to quantify uncertainty is indispensable.

[2]E. J. Gumbel, Statistics of Extreme, Dover Publication

The four rules of arithmetic to deal with uncertain data objects with fuzzy logic attributes are presented in section 4. Calculations with a fuzzy quantity is carried out in association with the upper cut-off, lower cut-off, an average and a spread of data. A false alarm is suspected when upper limit of an interval falls below its lower level.

3.2.7 Intelligent Sensor Network

Active Control: The sensor elements and the monitor types are programmable. The same sensor can be used to detect temperature, with a microphone attached, can also detect sound. Since the mechanisms are based on spectroscopy, which measures an input as a function of either wavelength or frequency, the focus can be programed to detect a certain chemical content in the air such as Anthrax. Since a micro-chip or a dedicated small computer is an integral part of the sensing/monitoring device, a direct evaluation of concentration of the bacteria Bacillus anthracis can be sent to the LiveDesignDataBase.

3.2.8 Sensors & Monitors

On-line Data: The raw response digital stream retrieved from sensor and monitor systems, can be exactly represented in terms of a collection of numbers. As an element of the intelligent sensor network adequate computing power is built to filter the data according to specific criteria that can be actively controlled by module number 7.

3.2.9 On-ground Experts' Action

Intervene: It is not prudent to completely rely upon an automated environment like the LiveDesign Emergency Management System, hence provision is made in module number 9 for a human expert to modify and override the instructions to the emergency victims via broadcast mechanism of module number 11.

In addition, if a shortcoming in the system is detected by the human expert, it is reported so that in the future it can be rectified in designing an Early Warning System.

3.2.10 Privacy Protection System Faults

Legal Issues: Sensing and monitoring devices, which harvest human data, comply with the following legal issues:

■ Protection of Privacy: In programming the intelligent sensor network the allowable limits for data collection should not be violated. But if, under some exceptional situation the emergency management authority takes any decision that breaches any restriction then the policy makers should be alerted to address such concerns.

■ ■ Product Liability: There will be some inevitable instance when injuries will be caused by some mal functioning of installed equipment. These failures are brought to the notice of the policy makers for future correction and dealing with the insurance issues arising out of such injuries.

3.2.11 Evacuation and Rescue

Broadcast: Common citizens will receive broadcast messages on how to leave the premises that has been declared by the authority to be under emergency. The civic authorities send trained and equipped first responders who enter the premise to rescue people out of danger. The broadcast messages are therefore very different for the rescuers and evacuees. Different broadcast follow through different channels of communications, such as loudspeaker or cell phone information directly to the evacuees but special purpose wireless devices for the first responders.

3.2.12 Technical and Administrative

Report: All actions taken by the emergency management team will be recorded using the LiveDesign Markup Language to be reviewed for future use and designing Early Warning Systems. Any on-line mitigation advancement by the human expert is also recorded and is used in the current phase of emergency mitigation.

3.3 Descriptive Blocks

3.3.20 Emergency Knowledge System

To avoid confusion, a superset of the LiveDesignDataBase is called the the LiveDesign Emergency Knowledge System. the collection of the LiveDesignDataBase alongside the response history of sensors and monitors reinforce by the broadcast and report from on the ground experts constitute the LiveDesign Emergency Knowledge System.

3.3.30 Data Collection System

3.3.40 Data Processing System

Numerical operations the current knowledge base

3.3.50 Communication Infrastructure

3.4 Three Unique Features

3.4.A Intelligent Sensor Network

3.4.A.1 Objective

Conventional sensor monitor systems output a large volume of data at a very high time rate. The intelligent sensor network is the mechanism to extract only the small subset of useful data associated with the context of the particular emergency.

3.4.A.2 Different Constituent Modules

Working in tandem, modules number 4,5, 7, 8 9 and 12 form the intelligent sensor network system. Individual sensor and monitor sends the raw data to a local programmable computer processor. All the inputs from different sensors and monitors are synthesized. The input are filtered according to fuzzy logic specification. These specifications come from the active control module number 7.

3.4.B Elimination of False Alarm

3.4.B.1 Objective

In each cycle through the LiveDesign computer program calculations the uncertainty coming from each piece of information accumulate via the arithmetic operations of addition, subtraction, multiplication and division. In common calculation procedures these errors overwhelm computing resources and results become error-prone hence unusable. Filtering each result of an arithmetic operation the level of uncertainty is kept under control. This ensures high degree of reliability of LiveDesign computations.

Since in conventional fuzzy arithmetic calculations, where the mean and spread of data is related to the interval, accumulation of errors causes many false alarm. This is a reason why emergency management authorities tend to decide all mitigation measures using their subjective judgement. This shortcoming of conventional procedures are circumvented using a special algorithm, which is included in section 4, written in a computer algebra language for numerical analysts to follow. It is anticipated that users will write this algorithm in their own computer program language of choice.

3.4.B.2 Different Constituent Modules

The main task of elimination of false alarms is carried out in module number 6. The filter based on a prescribed arithmetic tolerance is constructed in this module. In addition to screening out false alarms, module number 6, sends a fuzzy control algorithm, whose parameters are based on the eliminated false alarms, to module number 7. This accelerates the process of eliminating false alarms since the operations are carried out at the hardware level.

3.4.C Early Warning System

3.4.C.1 Objective

Normal engineering installations are routinely over-designed in order to strengthen against uncertain events. The LiveDesign management system will be triggered only when some intervention from the local or higher civic authorities are necessary.

Based on experiences Early Warning Systems can be designed for future use. The calibration will address three types of emergencies addressed under module number 5. So for most of the time the LiveDesign emergency management system will not process real-world on-line data. During this so called idle time the system will create scenarios of possible emergency situations as appropriate to mitigate natural, man-made or industrial disasters. The emergency management authorities will be informed of such so called yet to apprehended events.

Mathematical formulations, *e.g.* based on optimization theory Coupled with *exhaustive searches* and information propagation, it may be possible to discover the 'holes' in the present management strategies. Improvements in future design and briefing the lawmakers on such possible threats, are additional benefits for implementing an Early Warning System in conjunction with LiveDesign emergency management.

3.4.C.2 Different Constituent Modules

The lessons learned from dealing with an emergency is encapsulated within the knowledge base written in the LiveDesign Markup Language. The shortcomings of the technical deficiencies that amplified the negative impact are stored in the module number 2, labeled Infrastructure. Any legal and or privacy issues that caused the emergency or hampered better mitigation is resolved via experts' action and are stored within the future Policy module number 3. To what extent the system deficiencies may amplify future disaster will be evaluated on the basis of the information in module number 12 and 2.

The same physical installation of the sensor and monitor systems should function under all three types, natural, terrorist, industrial calamities. Hence it is essential to design the sensor and monitor systems following the intelligent sensor network paradigm.

4 Some Items in LiveDesign Computation

The following three items are elaborated in this section:

1. LiveDesignArithmetic
2. Detection of False Alarms and System Failure situations
3. Calibrating and testing Early Warning System

4.1 LiveDesignArithmetic

Arithmetic operations on uncertain variables, which are represented by exact numbers, is the main computational task in LiveDesign computation. An uncertain value x is estimated to lie between a and b:

$$a \leq x \leq b \tag{4.1}$$

This means that an uncertain variable x can take any number between a and b. Using the *interval arithmetic* paradigm, all numbers, which will be represented by X, between a and b are housed in `Interval`$[a, b]$. For example, if the variable x, which represents the possibility of catching fire, lies between 30% and 40% then in calculation X will be `Interval`$[.3, .4]$. Of course this inexactness gets compounded with each arithmetic operation leading to the fact that:

$$\text{in ordinary numerical calculations: } x \times (y + z) = x \times y + x \times z \tag{4.2}$$

$$\text{But in interval arithmetic: } X(Y + Z) \tag{4.3}$$

$$\text{is a subset of } XY + XZ \tag{4.4}$$

In other words, some information of equation (4.4) is missing in equation (4.3). As a result calculation error is created and gets magnified after each of the four operations of arithmetic. Of course, if $a < 0, b > 0$ the 0 is included in the interval hence the inverse of

$$\texttt{Interval}[a < 0, b > 0] \tag{4.5}$$

is undefined. In calculations such a division by zero is indeed a *fatal error.* Hence once the interval gets in the form equation (4.5) a false alarm is detected in solving an equation that involves division. This LiveDesignArithmetic provides a work around the theoretical shortcoming in the light of *security engineering*.

4.1.1 Symbolic Computation

G. William Walster, Ramon E. Moore and Eldon R. Hansen have 22 US patents (between 2002 – 04) related to *interval arithmetic*. Program modules in `C`$^{++}$ are available in: `www.boost.org/doc/libs/release/libs/numeric/interval/doc/interval.htm` The computing environment *Mathematica*, `http://www.wolfram.com/`, has the following operations built-in:

$$\texttt{Interval}[a, b] + \texttt{Interval}[c, d] = \texttt{Interval}[a + c, b + d] \tag{4.6}$$

$$\texttt{Interval}[a, b] - \texttt{Interval}[c, d] = \texttt{Interval}[a - d, b - c] \tag{4.7}$$

whereas:

$$\texttt{Interval}[a, b] \times \texttt{Interval}[c, d]$$
$$= \texttt{Interval}[min(a \times c, a \times d, b \times c, b \times d), max(a \times c, a \times d, b \times c, b \times d)] \tag{4.8}$$
$$\texttt{Interval}[a, b]/\texttt{Interval}[c, d]$$
$$= \texttt{Interval}[min(a/c, a/d, b/c, b/d), max(a/c, a/d, b/c, b/d)] \tag{4.9}$$

Within the context of fuzzy logic, *interval arithmetic* will be used for the membership functions, a $\beta-$distribution is used to reflect the skewness. This is called type-2 fuzziness some very powerful computational tools are available in Here a more robust numerical procedure is furnished harnessing properties of the $\beta-$functions.

4.1.2 Fuzzy Logic

Apart from the strict statements $\in [x]$ and $x \notin [x]$, intermediate values are also possible, to which real numbers $\mu \in [0, 1]$ are assigned, $\mu = 1$ corresponds to definite membership while $\mu = 0$ is non-membership.

4.1.3 Statistical mean and standard deviation calculations

Since two values in a complex computation can come from many independent system parameters, an efficient and engineering assumption with acceptable level of computational error will be to ignore the correlation between two values. This assumption of variable independency will add error in output values but standard algorithmic checks are implemented to ascertain that the calculation errors are not getting out of hands.

Thus any two variables x and y, will be assumed to be statistically independent for the purpose of the following arithmetic operations.

4.1.3.1 Addition and subtraction

$$\text{for two numbers: } z = x \pm y \tag{4.10}$$

$$\text{for their means: } \bar{z} = \bar{x} \pm \bar{y} \tag{4.11}$$

$$\text{for their standard deviations: } \sigma_z = \sqrt{\sigma_x^2 + \sigma_y^2} \tag{4.12}$$

4.1.3.2 Multiplication

$$\text{for two numbers: } z = x * y \tag{4.13}$$

$$\text{for their means: } \bar{z} = \bar{x} * \bar{y} \tag{4.14}$$

4.1.3.3 Inversion

$$\text{for a number: } z = \frac{1}{x} \tag{4.15}$$

in order to evaluate $\bar{z}, \sigma_z$, a $\beta-$distribution is assumed with appropriate parameters

4.1.4 Uncertainties represented within an Interval with Expected Value and Dispersion

By assuming a statistical distribution between the lower ad the upper bounds the expected value and dispersion of values can be recognized as the mean and standard deviation of the assumed distribution.

Each variable is represented by a quadruplet:

$$x \rightarrow \{a_x, b_x, \mu_x, \sigma_x\}; \quad \text{or } x \rightarrow \{a_x, b_x, \bar{x}, \sigma_x\}; \mu_x = \bar{x} : \text{mean}; \tag{4.16}$$

$$\sigma_x : \text{standard deviation}; \quad a_x : \text{lower cut-off} \quad ; b_x : \text{upper cut-off} \tag{4.17}$$

as explained below

4.1.5 Rules on data

$$\alpha = \{\texttt{Interval}[a_\alpha, b_\alpha], \bar{\alpha}, \sigma_\alpha\} \text{ and } \beta = \left\{\texttt{Interval}[a_\beta, b_\beta], \bar{\beta}, \sigma_\beta\right\} \tag{4.18}$$

$$\alpha + \beta = \left\{(\texttt{Interval}[a_\alpha, b_\alpha] + \texttt{Interval}[a_\beta, b_\beta)], (\bar{\alpha} + \bar{\beta}), \sqrt{\sigma_\alpha^2 + \sigma_\beta^2}\right\} \tag{4.19}$$

$$\alpha \pm \beta = \left\{(\texttt{Interval}[a_\alpha, b_\alpha] \pm \texttt{Interval}[a_\beta, b_\beta)], (\bar{\alpha} \pm \bar{\beta}), \sqrt{\sigma_\alpha^2 + \sigma_\beta^2}\right\} \tag{4.20}$$

$$\alpha * \beta = \left\{(\texttt{Interval}[a_\alpha, b_\alpha] * \texttt{Interval}[a_\beta, b_\beta)], (\bar{\alpha} * \bar{\beta}), \sqrt{\sigma_\alpha^2 * \sigma_\beta^2 + \bar{\alpha}^2\sigma_\beta^2 + \bar{\beta}^2\sigma_\alpha^2}\right\} \tag{4.21}$$

4.1.6 New rule for Inversion when the `Interval` contains the zero

If the inverse of a value

$$x \rightarrow \{a_x, b_x, \mu_x, \sigma_x\}; \quad \text{such that: } a_x \times b_x < 0 \tag{4.22}$$

Then

$$\text{to compute } z = \frac{1}{x} \tag{4.23}$$

$$z \rightarrow \{a_z, b_z, \mu_z, \sigma_z\}; \quad \text{from } \{a_x, b_x, \mu_x, \sigma_x\} \tag{4.24}$$

is computed in the following way. Here:

$$a_z < 0 \quad \text{whereas} \quad b_z > 0 \tag{4.25}$$

It should be noted that for all cases

$$a_z < b_z \tag{4.26}$$

hence, equation (4.25) is the only possibility for the `Interval[`a_z, b_z`]` to contain zero.

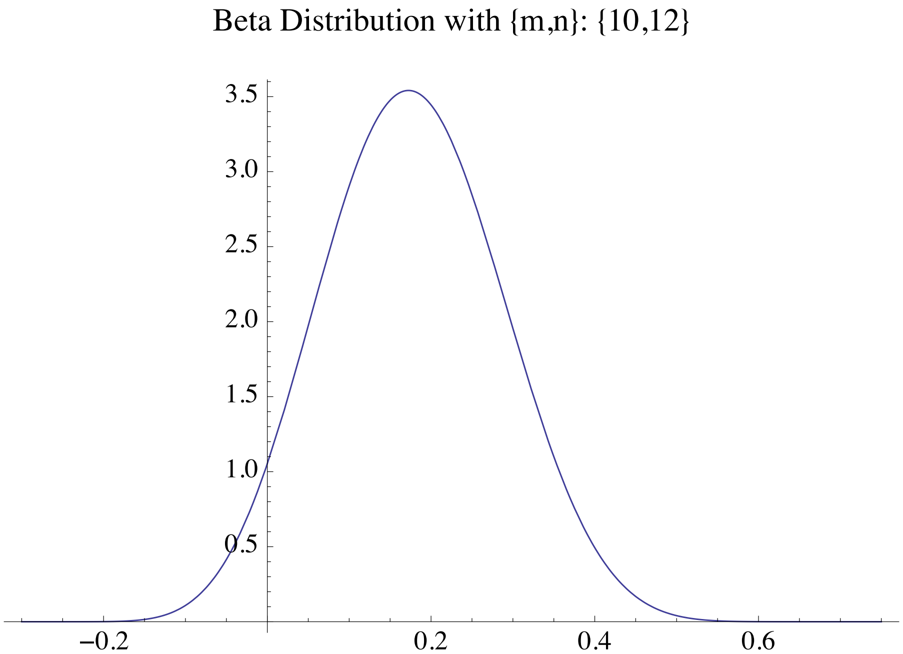


Figure 4.1: Uncertainty function $f(z)$ in z that lies between -.2 and .6

4.1.7 Preserving the mean and standard deviation

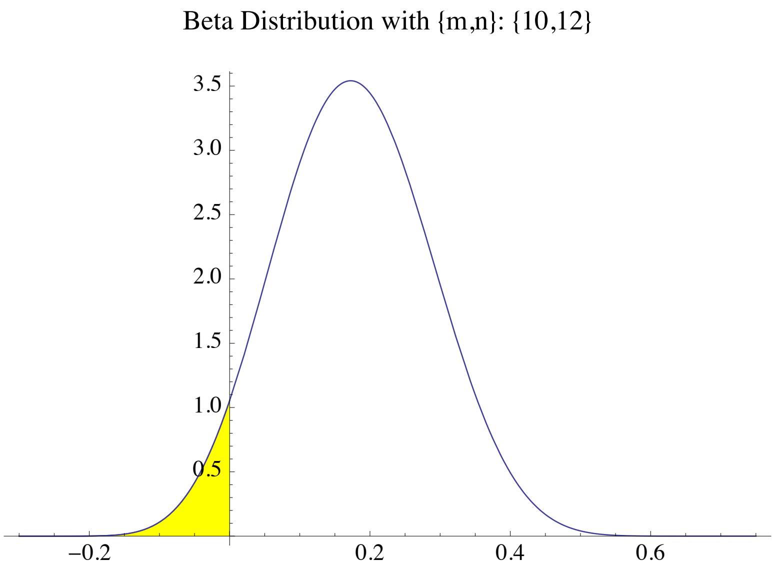


Figure 4.2: Area on the negative z region

The value of the yellow area in Figure 4.2 is

$$\int_a^0 f(z)\ dz = 0.0499047 \tag{4.27}$$

The cut-off for the modified $\beta-$function is $z_{cut-off}$ such that:

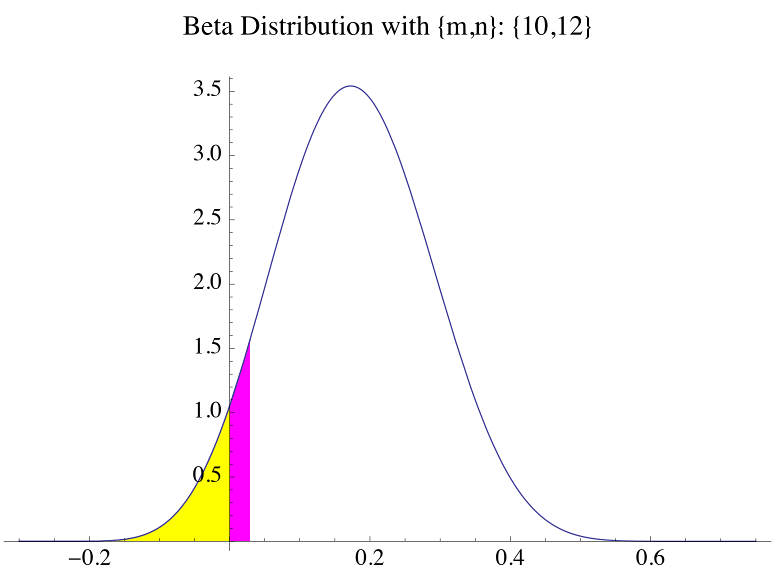


Figure 4.3: Equal areas on the negative and positive z regions

$$\int_a^0 f(z)\ dz = 0.0499047 = \int_0^{z_{cut-off}} f(z)\ dz \tag{4.28}$$

The $\beta-$function that has the same mean and standard deviation as in the original one shown in Figure 1.1 but between `Interval[`$z_{cut-off}, b$`]` is:

$$f_{modified}(z) = 12.7917(1 - 1.38705(x - 0.0290458))^{3.88011}(x - 0.0290458)^{0.263015} \tag{4.29}$$

this is shown below:

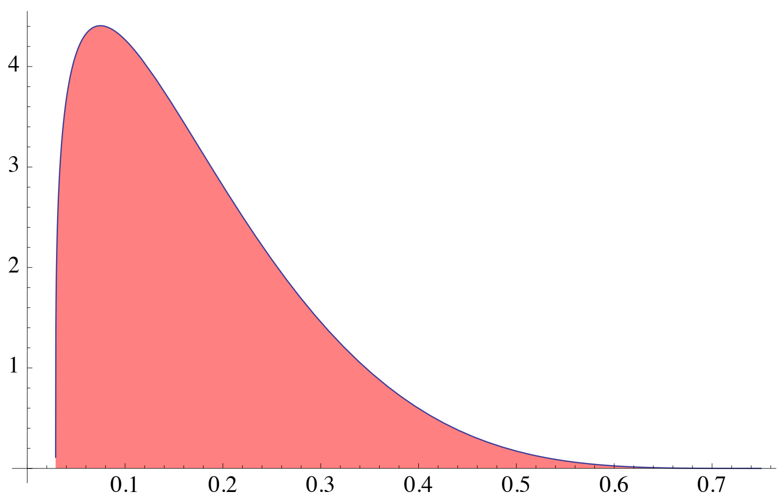


Figure 4.4: Modified uncertainty function with the same mean and standard deviation of the original $\beta-$function

The preservation of μ and σ is validated by:

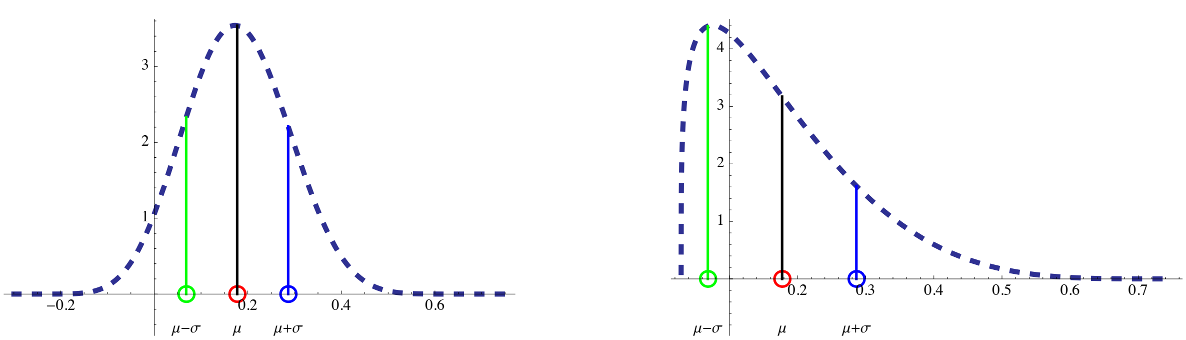


Figure 4.5: Left: original distribution; Right: modified distribution

Zero is not included within the interval of the modified distribution, hence its inverse can be well-defined using $f_{modified}(z)$ from equation (4.29). But in the spirit of equation (4.21):

$$z_{inverse} = \left\{ \texttt{Interval}[\frac{1}{b}, \frac{1}{z_{cut-off}}], \frac{1}{\mu_z}, \sigma_{inverse} \right\} \tag{4.30}$$

The response of the system can be classified into four broad categories.

(i) False alarm:

The system will always have some flaws that may trigger alarm based on some wrong input of overall computer program bugs. No action should be taken by the users but the management has to announce a false alarm.

(ii) Safety issue:

Some actions are needed b the local authority, for example, in-house safety officers, local police forces and fire brigades and medical personnel. These services are routinely provided in every big city by the civic authorities. Since they occur very frequently common statistics of mean and standard deviations are used to estimate the resources and budgets. Regular systematic drills are very common and the subjective information of the emergency management authorities is extremely reliable.

(iii) Security issues:

Severe natural disasters, like Katrina, industrial accidents, like the BP oil spill, and an-made disasters, like 9/11, demand a very different emergency response from the ones observed every day in big cities.

The infrequent occurrences of such catastrophic events prevent common statistical tool of means and standard deviations to be effective. A special branch of mathematics studies these cases under the topic *extreme statistics*.

The distinction between Safety issues and Security issues is further highlighted by the type of statistics one employs to predict mitigation. Commonly we say the intensity of the safety issue is around two o three standard deviation away whereas for a security issue we say it is so severe that it occurs once in 30 or 50 years. Commonly we refer to the DHS color coding to describe emergency. For example, Green and Blue alerts could fall under safety violation but Orange and Red alert could be characterized as a security breach. These are subjective decisions of experts. A precise demarcations in the color alert system is of little interest. This is indeed a fuzzy representation that the LiveDesign-computing engine employs extensively.

(iv) System failure:

Like any real-world automatic system any running LiveDesign operations will encounter hardware limitations and software bugs. During the calibration and testing an Early Warning System many such shortcomings will be fixed but under any severe emergency state additional care must be exercised by human mentors who must be the final arbitrator of all decisions. From the mathematical operations, such as division by zero, some system limitations can be detected immediately.

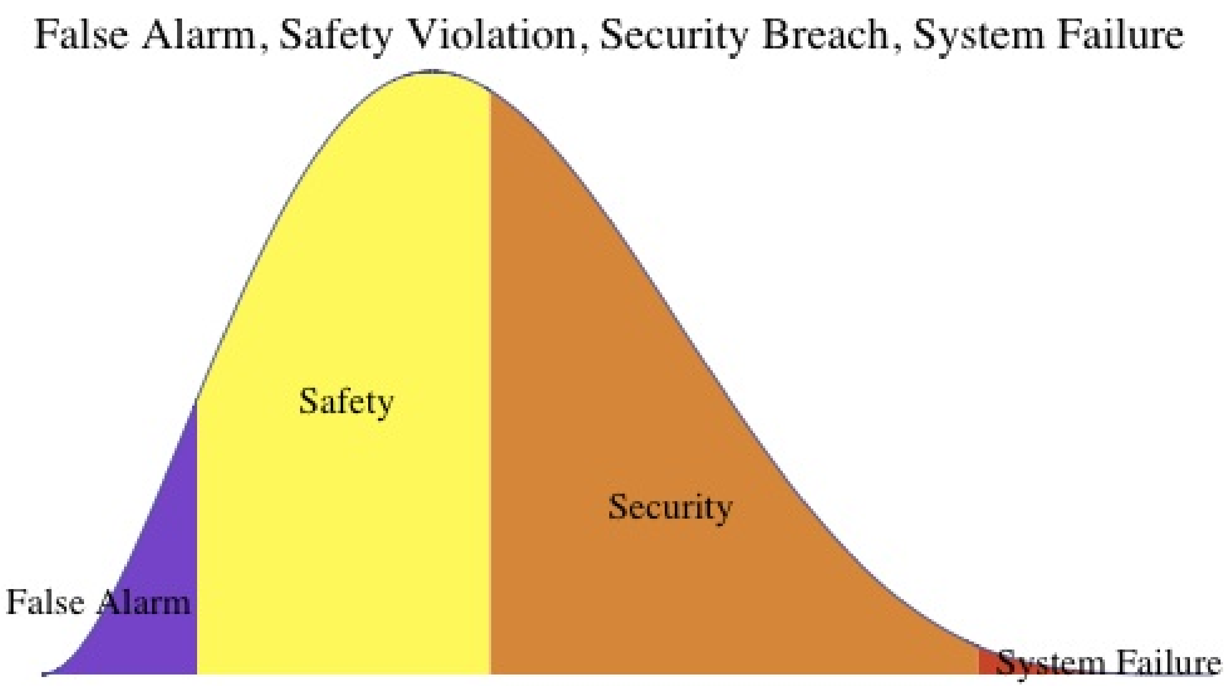


Figure 4.6: Severity and typical participation of LiveDesign outcomes

In order to facilitate communication a color coding within the safety and security region in Figure 4.6 can be implemented. Alternatively, a Richter scale type an emergency scale can used to communicate the severity of the emergency. Fuzzy Logic is the best mathematical tool to handle these subjective interpretation of LiveDesign computation outcomes.

The following computing characterizations are built into the LiveDesign engine to classify a certain outcome to fall within the four segments shown in Figure 4.6.

4.1.8 Detection of a False Alarm

In Figure 1.1 module number 11 receives the broadcast information based on the on-line data received in module number 8 via the module number 6 that decides, what data to request the intelligent sensor network and what is the meaning of the composite picture from the input data. All these information is organized in a correlation matrix. Any inconsistency in the calculated value of a correlation coefficient will inform the emergency management authority that a false alarm is encountered. An example follows.

Suppose two chemicals are detected by the intelligent sensor network. Suppose the LiveDesignDataBase has the information that the observed concentrations of such offending chemicals will ignite fire and emit a certain type of smoke. Now the intelligent sensor network is instructed to detect and measure the temperature distribution in that room and measure the concentration of the type of smoke apprehended. Now, from the on-line data collection if no high temperature nor smoke is found then the correlation between the apprehended smoke concentration and the actual one is nearly zero. The same is the case for the temperature distribution. This is commonly detected as a false alarm. Now the emergency management authority and the system engineers will address this issue while the disaster is unfolding. Many such cases are fixed during the dry runs.

In addition, the on-ground experts, indicated in Figure 1.1 module number 9, may find no fire nor any smoke, when the system failed to detect this false alarm, and will intervene through module number 12 where a Bayesian statistical correction will be implemented on-line and any future reoccurrence will be averted.

4.1.9 Detection of a System Failure

Since LiveDesign system will be used in real-time to mitigate extreme disasters like Katrina, BP oil spill and 9/11, it should be kept in mind a "yet to be apprehended disaster" may overwhelm the system. Of course the human mentors as the emergency management authority will readily detect some very gross system limitation. However, a number of significantly small mis-judgements may culminate to a greater misery. Many such faults can be detect mathematically, as the LiveDesign system is running. Now, such situations are addressed.

4.1.10 Analytical Detection of a False Alarm and a System Failure

4.1.10.1 Basic mathematical form of an Emergency Management System

Among many variables, the experts along with the computer system engineer will select V number of variables. They will encompass V_s number of *subjective* variables and V_o number of *objective* variables. All V_s number of *subjective* variables will be quantified via fuzzy logic conversion of textual opinion into numbers and store in a variable-object z_i:

$$Z_I : \{z_i, [a_{z_i}, b_{z_i}], (\mu_{z_i}, \sigma_{z_i}), \{m_{z_i}, n_{z_i}\}\} \tag{4.31}$$

Since all *objective* variables are also stored in the above format of equation (4.31), the four rules of arithmetic described in §4.1 can be applied to all variables.

The threat indicators, all λ_i, are calculated as eigenvalues of a positive definite system $[\mathcal{M}]$:

$$[\mathcal{M}] \quad \text{a positive definite matrix of size } \Lambda \times \Lambda \tag{4.32}$$

The formulation of the threat matrix $[\mathcal{M}]$, which depends on the requirements from the emergency management authority, is an integral part of the software design aspect of the particular LiveDesign-engine. The positive definiteness of $[\mathcal{M}]$ assures all threat indicators, λ_i, will yield a positive value:

$$\lambda_i > 0, \text{for all } i \tag{4.33}$$

(like a threat thermometer or a threat barometer) when the LiveDesign-engine detects a threat. Keeping the options that some threat indicators will not be triggered in a disaster, i.e., all threat indicators will not be triggered in all disasters, equation (4.32) is relaxed to

$$[\mathcal{M}] \quad \text{a positive semi-definite matrix of size } \Lambda \times \Lambda \tag{4.34}$$

$$\text{when } \lambda_i \geq 0; \qquad \text{when } \lambda_i < 0 : \text{noise overshadowing decision} \tag{4.35}$$

will indicate computational noise, which builds up after each arithmetic operation and then cascades through the entirety of future calculations. Also, indefinitely large λ_i indicates a system failure:

$$\text{when } \lambda_i \to \infty : \text{ system failure} \tag{4.36}$$

Since the calculations of eigenvalues λ_i involve solving a higher degree equation where square roots are inevitable, a λ_i has a possibility to be a complex number, then:

$$\lambda_i : \text{ complex, imaginary part of } \lambda_i \neq 0 \quad \text{indicates inconsistent system design} \tag{4.37}$$

4.1.11 An illustration

In order to emphasize the usefulness of the proposed LiveDesignArithmetic operations a 2 by 2 matrix calculation is now demonstrated.

a matrix of the form:$[a] = \begin{bmatrix} \alpha & \beta \\ \gamma & \delta \end{bmatrix}$ is randomly gnerated within Interval[{0, 1}] :

$$[a] = \begin{bmatrix} \text{Interval}[\{0.315022, 0.399975\}] & \text{Interval}[\{0.286956, 0.556781\}] \\ \text{Interval}[\{0.286956, 0.556781\}] & \text{Interval}[\{0.220422, 0.82961\}] \end{bmatrix} \tag{4.38}$$

computed $[b] = [a]^{Transpose}[a]$

$$[b] = \begin{bmatrix} \text{Interval}[\{0.181582, 0.469984\}] & \text{Interval}[\{0.278161, 0.770452\}] \\ \text{Interval}[\{0.278161, 0.770452\}] & \text{Interval}[\{0.513991, 1.28331\}] \end{bmatrix} \tag{4.39}$$

4.1.11.1 Eigenvalues as per Prior Art

The computer mathematics program *Mathematica* calculated the two eigenvalues as:

$$\left\{ \begin{array}{l} \frac{1}{2}(\text{Interval}[\{0.695573, 1.7533\}] - \text{Interval}[\{1.27249i, 2.2528\}]) \\ \frac{1}{2}(\text{Interval}[\{0.695573, 1.7533\}] + \text{Interval}[\{1.27249i, 2.2528\}]) \end{array} \right\} \tag{4.40}$$

Here the complex numbers appeared in the intervals, there is obviously something wrong!

4.1.11.2 Eigenvalues as per the LiveDesignArithmetic

The same calculations had the following square root term :

$$\sqrt{\left(-\alpha^2-\beta^2-\gamma^2-\delta^2\right)^2-4\left(\alpha^2\delta^2-2\alpha\beta\gamma\delta+\beta^2\gamma^2\right)} \tag{4.41}$$

$$\text{evaluated to:Interval}[\{0.532979, 1.88243\}] \text{ without any imaginary quantity} \tag{4.42}$$

During the first pass the eigenvalues were evaluated as:

$$\{\lambda_1, \lambda_2\} = \{\text{Interval}[\{-1.42399, 0.734616\}], \text{Interval}[\{0.48982, 2.64843\}]\} \tag{4.43}$$

The first threat criterion λ_1 exhibits a false alarm possibility, between `Interval[` $-1.42399, 0$`]` and the second threat criterion λ_2 exhibits a system failure possibility, between `Interval[`$2, 2.64843$`]` since from the theoretical view point threat criteria $\lambda_1, lambda_2$ must lie between `Interval[`$0, 2$`]`. The computation engine in module number 6 in Figure 1.1 eliminates the false alarm and the "Active Control" module number 7 in Figure 1.1 detects the system failure and these two actions update the LiveDesignDataBase leading to:

$$\{\lambda_1, \lambda_2\} = \{\text{Interval}[\{0, 0.734616\}], \text{Interval}[\{0.48982, 2\}]\} \tag{4.44}$$

that is used to carry out the next cycle of LiveDesign-computation for broadcast via module number 11 in Figure 1.1.

4.1.12 Numerical Detection of a System Failure

This originates from a condition, when a "division by zero," cannot be avoided. In a sense, Figure 4.3 becomes ineffective hence Figure 4.5 cannot be constructed. An example follows.

4.1.12.1 An example of failure to construct Z^{-1}

. Consider:

$$Z : \{z, [a_z, b_z], (\mu_z, \sigma_z), \{m_z, n_z\}\} = \{z, [-.45, .6], (0.0272727, 0.109017), \{10, 12\}\} \quad (4.45)$$

The associated locations of the cut-offs and mean and the extent of the standard deviation is shown below in Figure 4.2.

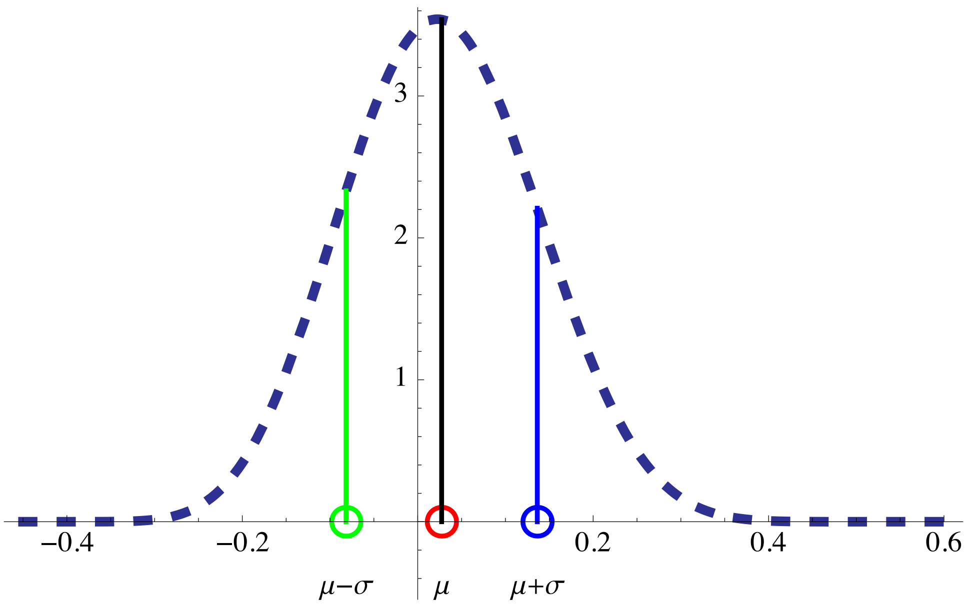


Figure 4.7: An example with the same shape as in Figure 4.5

For this case still the region $x < 0$ is less than the one for $x > 0$. Applying the unbiased argument that equal error should be considered on both sides one gets the following Figure 4.8: where the mean lies within the region of error, thus it will be too careless to think that the calculation so far lied within any useful error bound. The steps indicated in equations (5.29) through (5.37) become useless.

This fatal error must be detected and report as the system most likely would yield very erroneous results.

Hence, a system failure has been numerically detected.

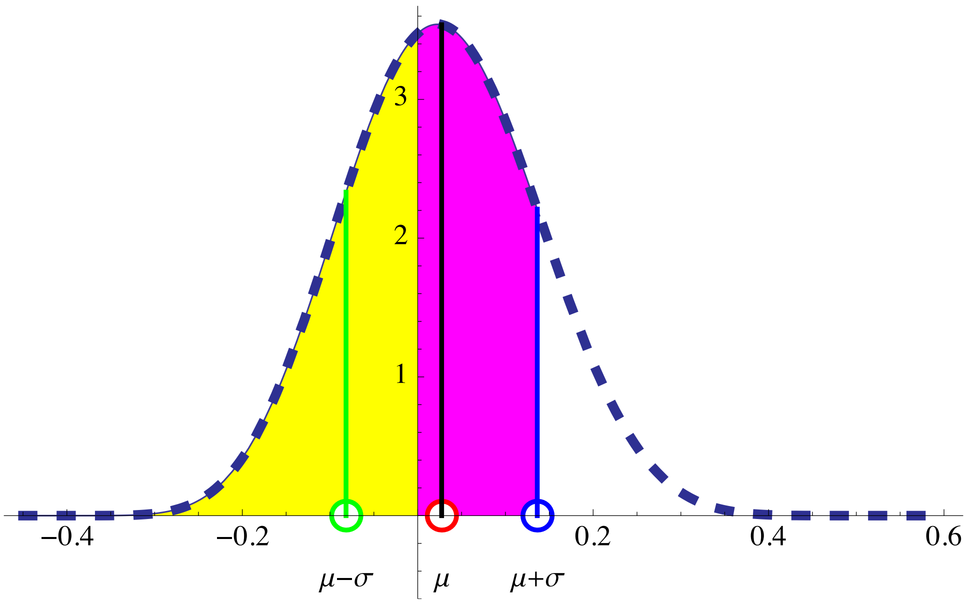


Figure 4.8: Unlike in Figure 4.3 the mean lies within the region of error

4.2 Calibrating and testing Early Warning System

The efficiency of the LiveDesign software-hardware integration is evidence by its capabilities to discover future disasters and prepare the emergency management authority to train the first responders and make the citizens aware of potential threats. The key steps in this phase of executing the LiveDesign computational engine are described anticipating parallel computing environment. Hence instead of typical flow charts, which signify serial computing, Figure 4.9 and Figure 4.10 show how different computing resources collectively act.

4.2.1 Updating

show the connection of the satellite computing units with respective central computing units.

(A) in Figure 1.1, as encountered in a real situation

(1) module number 12 reports the engineering design flaws to module number 2

(2) module number 10 reports the policy shortcomings to module number 3

and enriches the existing engineering and policy information in module number 1 based on the new observation:

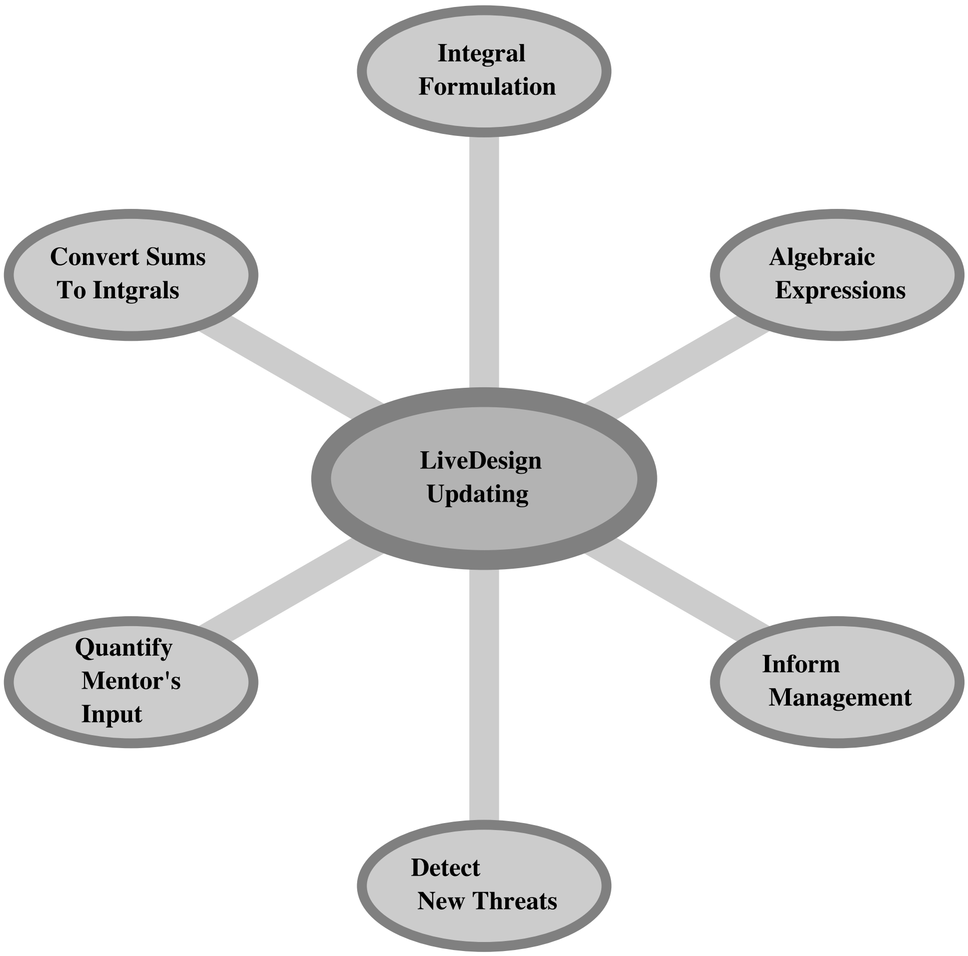


Figure 4.9: Updating the engineering requirements and policy needs

The new aspects are indicated in Figure 4.9. The new aspects are described as follows:

4.2.1.1 LiveDesignUpdating

This is the basic Bayesian computation engine. It can handle both algebraic and integral form of the Bayesian Theorem.

4.2.1.2 Quantify Mentor's Input

A Fuzzy Logic computer program will quantify human expert's subjective opinions, the LiveDesign Markup Language will directly translate the recommended changes in the engineering specifications to update the LiveDesignDataBase.

4.2.1.3 Convert Sums to Integrals

For a large number of tightly packed discrete variables x_i a sum can be converted into an equivalent integral:

$$\text{for arbitrary } \varphi; \quad \sum_i \varphi(x_i)(x_{i+1} - x_i) \approx \int \varphi(x)dx \tag{4.46}$$

Since many integrals can be evaluated exactly using computer algebra systems there is a 1000 folds or more saving in computing time when evaluation of large sums are avoided.

4.2.1.4 Integral Formulation

For continuous variables, *e.g.* temperature, smoke density, concentration of offending chemicals, the integral form of the Bayesian theorem is suitable.

4.2.1.5 Algebraic Expressions

For discrete variables, *e.g.* number of rooms in a floor, number of exits for the occupants to escape, number of emergency doors available to the first responders, the algebraic form of the Bayesian theorem is suitable.

4.2.1.6 Inform Management

The management receives the most up to date information about the disaster state from the LiveDesign system. In addition to textual communications, graphic depictions of the current scenario is a crucial aspect for efficient mitigation. Transmission of graphic information is elaborated in §4.2.3.

4.2.1.7 Detect New Threats

(B) in Figure 1.1, as encountered in a real situation enriches the existing policy information set with this new item;

(C) the intelligent sensor network can be tested as a separate stand-alone system in a technical laboratory since adequate built-in computing power is an integral part of the intelligent sensor network running the same software environment furnished by the LiveDesign Markup Language module number 1 of Figure 1.1;

(D) the dry run simulations can be carried out:

- (i) by employing the conventional reliability-based statistical formulations, and computing packages to investigate the safety violations, refer to Figure 4.6 to demarkate between safety and security. This is a well established topic and will not be discussed here.
- (ii) by employing the mathematics and numerics of the *Extreme value statistics* to investigate the security breaches for the following cases:
  - (a) Type-I: Natural Calamity — Limited applications can be found in the literature. Detailed mathematical development is available in
  - (b) Type-II: Terrorist Attacks — This is completely new. The Game Theory calculation is carried out in tandem to foil possible terrorist moves.
  - (c) Type-III: Industrial Disasters — Limited applications can be found in the literature. Detailed mathematical development is available in

4.2.2 Dry runs and Calibrations

The new aspects included in this simulation are summarized below.

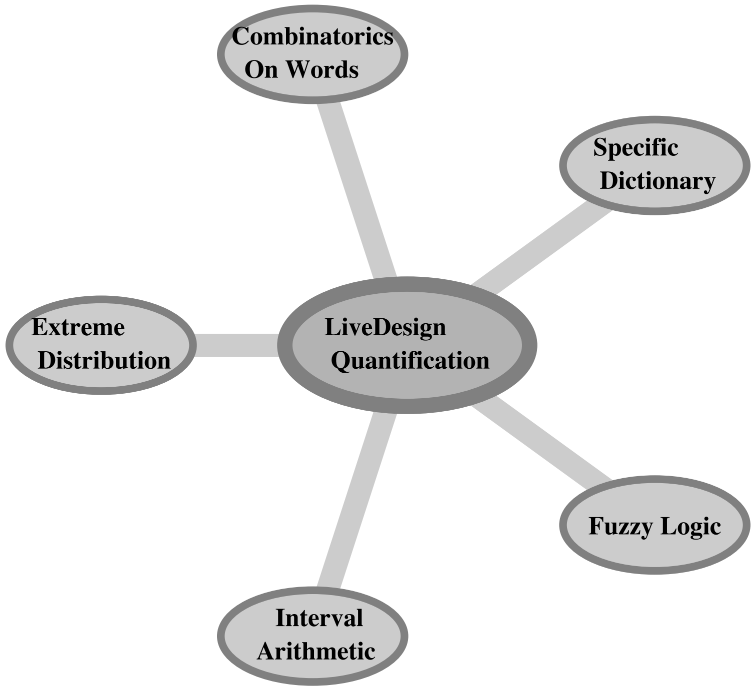


Figure 4.10: Threat Simulation

4.2.2.1 LiveDesign Quantification for Broadcast

Textual broadcasting needs translation from LiveDesignDataBase into human understandable linguistic form. For example, for a certain concentration of smoke and offending chemical the broadcast message should say something like: "leave the building in 45 seconds."

The graphical broadcast needs rending images with text labels. Some examples are inluded in §4.2.3.

4.2.2.2 Extreme Distributions

The safety related threat distributions are the familiar bell shaped curves:

$$\text{normal distribution} : f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \tag{4.47}$$

$$\text{extreme value distribution Type-I} : \frac{e^{-\frac{x-\mu}{\sigma}}}{\sigma} \left(e^{-e^{-\frac{x-\mu}{\sigma}}}\right) \tag{4.48}$$

whereas all three types of extreme value distributions are very different. The exponent of square type, equation (4.47) is used for simulating safety issues whereas the double exponent, equation (4.48), will be used in simulating a massive natural disaster. Unlike normal distributions, for security breaches, Figure 4.6, the mean nor standard deviation has any intuitive meaning. In the numerical simulations for extreme cases the frequency $f_x(x)$ as the double exponential,*e.g.* equation (4.48), is also of little direct value. The cumulative distribution function:

$$\wp(x) = \int_{-\infty}^{x} f_x(x)dx \qquad (4.49)$$

is to be used in order to simulate cases of extreme natural calamities, industrial disasters and terrorist attacks. All three extreme value distributions of Type-I, Type-II and Type-III, share the following common form:

$$\wp(x)\Big|_{\mu,\sigma,\xi} = e^{-\left[1+\xi\left(\frac{x-\mu}{\sigma}\right)\right]^{-\frac{1}{\xi}}} \qquad (4.50)$$

$$\mu : \text{ the location parameter}$$

$$\xi \in \Re : \text{ the shape parameter}$$

4.2.2.3 Combinatorics on Words

The selection of x values of equation (4.50) in a sample can be carried out by employing this novel technique. Generation of extreme samples is not common. Popular random number generators are suitable for safety modeling where the mean driven statistics play the significant role because an experiment, *e.g.* tossing of an unbiased coin or rolling an unbiased disc, can be conducted a large number of times. Events like Katrina nor 9/11 are o uncommon that random number generators defeat the purpose. Based on the theoretical computer science branch of “Combinatorics on Words,” shows that there are certain patterns that could be realized only once in a million or more experiments.

Associating a fuzzy logic membership function on a partially successful pattern in the four color chain of Figure 4.11 once can obtain extreme cases of various degrees.

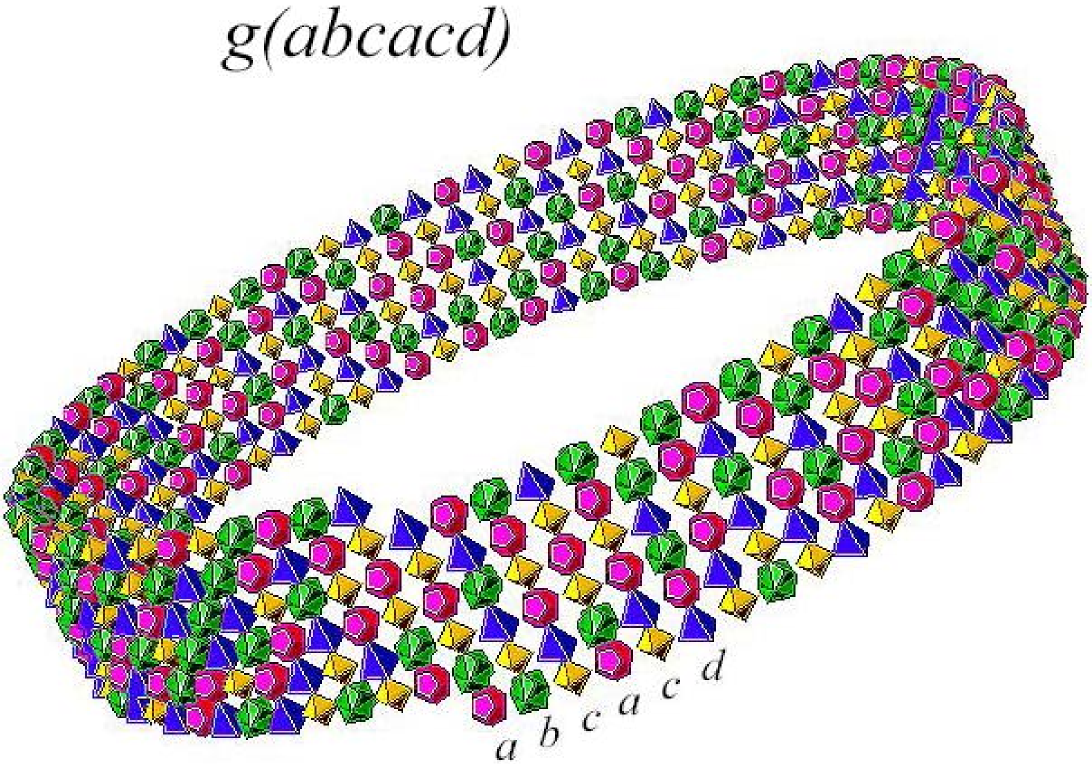


Figure 4.11: An Example of a Keränen Chain: in search of an Abelian square-free pattern

Using ordinary random number generators such possibilities do not exist.

4.2.2.4 Specific Dictionary — Adding rules during the run time

The LiveDesign Markup Language is written along side a set of (usually `Unix`) programs to manipulate the LiveDesignDataBase. Direct input, *e.g.* via keyboards, voice, hand gesture, are possible. Mathematical operations involve Bayesian calculations,

4.2.2.5 Fuzzy Logic on Game Theory constructs and Bayesian updating

The Game Theory pay-off matrices will have *fuzzy numbers,* which can mostbe manipulated by rules stated in §4.1.5 and §5.1.

4.2.2.6 Interval Arithmetic

This issue in general is described in details in , and will not be repeated here. All arithmetic rules are also stated in §4.1.5 and §5.1.

4.2.3 Threat Simulation and Data Compression

A new way of broadcasting large graphics outputs make it possible for the LiveDesign engine to be effective under extreme disaster situations.

As technology is growing new display devices are appearing in the consumer electronic market when the need to broadcast more detail graphical instructions are becoming critical in order to meet user expectations.

The mathematical problem is to break-down a picture into convex chunks, provides the tools to send graphics to 3-D TV monitors, and in particular, formulations from can be used for 2-D cases, *e.g.* that involves floor plans. For 3-D TVs formulations in can be employed using the

A TV monitor with 1920 by 1080 pixels needs 6,220,800 RGB pixel values: where as 500 polygon pieces of average 6 sides:

$$\text{conventional file size} : 6,220,800 \tag{4.51}$$

$$\text{compressed file size} : 9000 \tag{4.52}$$

$$\text{proposed compression advantage is almost} : 700 \text{ times} \tag{4.53}$$

may be able to describe the figure when the outlines are algorithmically extracted. Now using interpolations within each polygon piece, the graphics can be created after only 9000 RGB polygonal vertex values are broadcast. As a part of the intelligent sensor network the LiveDesign hardware integration meets this computational demand.

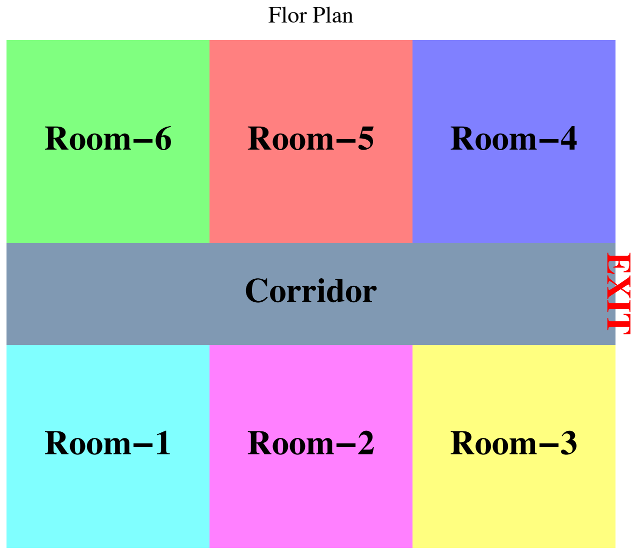


Figure 4.12: Floor Plan stored in LiveDesignDataBase

For example, disaster in a six room floor, Figure 4.12, is demonstrated to test a simulated case of fire, smoke and chemical concentration.The LiveDesignDataBase has the information that emergency doors will open or close depending on the situation, in order to stop spreading the combined hazard. From the numerical values the temperature and smoke and chemical concentrations, experts have made the estimation of the hazard available in module number 1 of Figure 1.1.

Now as the disaster is unfolding the threat contours are continuously updated by refreshing the graphics on monitors.

All three types of disasters, *e.g.* natural, industrial and terrorism, can be simulated using the data compression, which is elaborated here, according to equation (4.50).

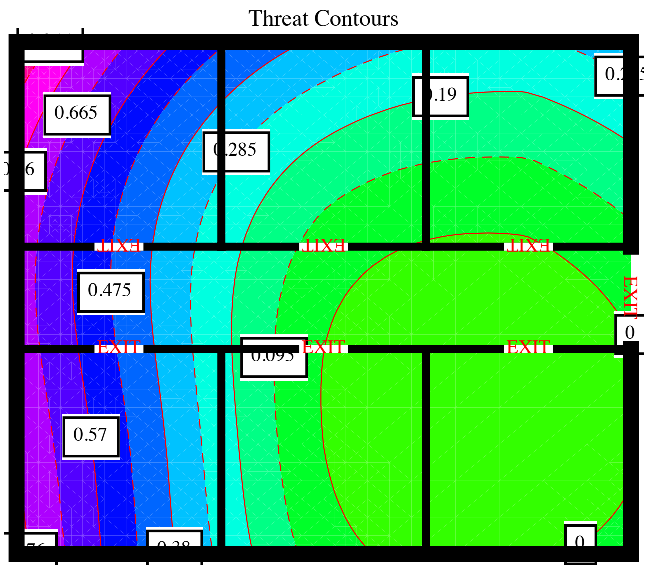


Figure 4.13: First time detected Threat: as threat contours

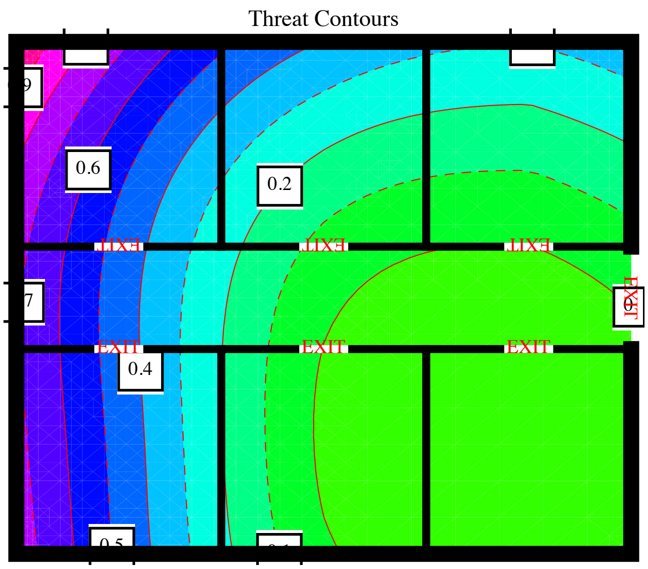


Figure 4.14: Current level of Threat: as threat contours

5 Mathematical Development and Proofs

5.1 Inversion of numbers with `Interval Arithmetic`

5.1.1 Initial function: represented as a function object

$$f_x(x) = \frac{x^{(m-1)}(1-x)^{(n-1)}}{B[m,n]} \tag{5.1}$$

$$f : \{\beta, x, \{m,n\}, \{0,1\}\} \text{ — represented as an object} \tag{5.2}$$

The mean and standard deviation, μ and σ, respectively, are calculated from m and n using the closed-form expressions:

$$\mu_x = \frac{m}{m+n}; \qquad \sigma_x = \sqrt{\frac{mn}{(m+n)^2(m+n+1)}} \tag{5.3}$$

$$\text{conversely: } m = \frac{-\mu^3+\mu^2-\mu\sigma^2}{\sigma^2}; \qquad n = \frac{\mu^3-2\mu^2+\mu\sigma^2+\mu-\sigma^2}{\sigma^2} \tag{5.4}$$

It is to be noted that, using the scaling of $(b-a)$ and shift of a:

$$\text{when } a<x<b; \quad \mu_x = a + \frac{m(b-a)}{m+n}; \qquad \sigma_x = (b-a)\sqrt{\frac{mn}{(m+n)^2(m+n+1)}} \tag{5.5}$$

$$\text{because } f_x(x) = \frac{\left(\frac{(x-a)}{(b-a)}\right)^{(m-1)}\left(1-\frac{(x-a)}{(b-a)}\right)^{(n-1)}}{B[m,n]} \tag{5.6}$$

5.1.2 Evaluation of $f_y(y)$

We would like to compute the mean, μ_y, and standard deviation σ_y and those indices m, n of y when:

$$y = -x \text{ and } \frac{1}{x}; \text{ generic arithmetic additive and multiplcative inverses} \tag{5.7}$$

We start with the "conservation of the *positive* probability mass" within each pair of differential elements:

$$f_y(y)dy = f_x(x)dx \rightarrow f_y(y) = f_x(x)\left\|\frac{dx}{dy}\right\| \tag{5.8}$$

5.1.3 Additive inverse with respect to the additive neutral 0

$$y = -x; \qquad \left\|\frac{dx}{dy}\right\| = \left\|\frac{dy}{dx}\right\| = 1 \tag{5.9}$$

$$f_y(y) = \frac{(-y)^{(m-1)}(1+y)^{(n-1)}}{B[m,n]} \tag{5.10}$$

$$\text{as a function object } \to f : \{\beta, y, \{n,m\}\{-1,0\}\} \tag{5.11}$$

5.1.4 Multiplicative inverse with respect to the multiplicative neutral 1

$$= f_x(x)\ \left\|-\frac{1}{y^2}\right\| \tag{5.12}$$

$$= f_x(x)\ \left(\frac{1}{y^2}\right) = x^2 f_x(x) = \frac{x^{(m+1)}(1-x)^{(n-1)}}{B[m,n]} \tag{5.13}$$

$$= \frac{\left(\frac{1}{y}\right)^{(m+1)} \left(1-\frac{1}{y}\right)^{(n-1)}}{B[m,n]} \tag{5.14}$$

5.1.5 Integration

To use standard results it is convenient to change the variable of y to x.

5.1.6 Mean

$$\mu_y = \int_1^\infty y\ f_y(y)\ dy = \int_0^1 \left(\frac{f_x(x)}{x}\right)\ dx = \int_0^1 \frac{x^{(m-2)}(1-x)^{(n-1)}}{B[m,n]}\ dx \tag{5.15}$$

$$= \frac{B[m-1,n]}{B[m,n]} = \frac{m+n-1}{m-1} \tag{5.16}$$

5.1.7 Standard deviation

$$s_y^2 = \int_1^\infty y^2\ f_y(y)\ dy = \int_0^1 \left(\frac{f_x(x)}{x^2}\right)\ dx = \int_0^1 \frac{x^{(m-3)}(1-x)^{(n-1)}}{B[m,n]}\ dx \tag{5.17}$$

$$= \frac{B[m-2,n]}{B[m,n]} = \frac{(m+n-1)(m+n-2)}{(m-1)(m-2)} \tag{5.18}$$

$$\text{or } \sigma_y^2 = s_y^2 - \mu_y^2 = \frac{mn+n^2-n}{(m-2)(m-1)^2} \tag{5.19}$$

$$\text{or } \sigma_y = \left(\frac{1}{\|m-1\|}\right)\sqrt{\frac{n(m+n-1)}{(m-2)}} \tag{5.20}$$

5.1.8 Calculations with Intervals

To account for uncertainties in a generic varible Z, its current value z, which lies between the lower and upper-offs, a_z, b_z, respectively, is represented using Interval Arithmetic. In addition to the interval, $\{a_z, b_z\}$, a function is employed to describe the behavior between those two cut-off ends in a statistical sense using stipulated mean and standard deviation, μ_z, σ_z, respectively. Using these four pieces of information, a_z, b_z, μ_z and σ_z, a $\beta-$ function as in equation (5.1), can uniquely characterize the behavior of Z. Furthermore, other statistics, like the skewness and kurtosis, can be estimated following this assumed $\beta-$function. Since the associated canonical $\beta-$ function, defined between 0 and 1 as in equation (5.1) involves two parameters, m, n, it is convenient to indicate the an object Z by:

$$Z : \{z, [a_z, b_z], (\mu_z, \sigma_z), \{m_z, n_z\}\} \tag{5.21}$$

This, equation (5.21) is indeed an over-specification since:

$$\{\mu_z, \sigma_z\} \qquad \text{can be calculated from: } \{a_z, b_z\}, \{m_z, n_z\} \tag{5.22}$$

$$\{m_z, n_z\} \qquad \text{can be calculated from: } \{a_z, b_z\}, \{\mu_z, \sigma_z\} \tag{5.23}$$

5.1.9 Inversion of a data object

There are two inversion operations, for subtraction and division, needed to carry out all four rules of arithmetic.

5.1.10 Subtraction from the structure

From equation (5.21):

$$-X : \{-x, [a_{-x}, b_{-x}], (\mu_{-x}, \sigma_{-x}), \{m_{-x}, n_{-x}\}\} \tag{5.24}$$

$$\text{is obtained from } X : \{x, [a_x, b_x], (\mu_x, \sigma_x), \{m_x, n_x\}\} \text{ as:} \tag{5.25}$$

$$a_{-x} = -b_x; \quad b_{-x} = -a_x; \quad \mu_{-x} = -\mu_x;; \quad \sigma_{-x} = \sigma_x; \quad m_{-x} = n_x; \quad n_{-x} = m_x \tag{5.26}$$

Also, as a special rule, in order to enforce:

$$X + (-X) = 0 : \{0, \{0, 0\}, \{0, 0\}, \{\phi, \phi\}\}; \tag{5.27}$$

equation (5.26) is coded explicitly.

5.1.11 Division structure

The multiplicative inverse of Z in the above equation (5.21) will be denoted by Z^{-1} provided. A significant retriction is that a_z and b_z, refer to equation (5.21), must be of the same sign:

$$a_z \times b_z > 0 \tag{5.28}$$

$$Z^{-1} : \{z^{-1}, [a_{z^{-1}}, b_{z^{-1}}], (\mu_{z^{-1}}, \sigma_{z^{-1}}), \{m_{z^{-1}}, n_{z^{-1}}\}\}; \tag{5.29}$$

$$\text{where: } z^{-1} = \frac{1}{z} \tag{5.30}$$

$$a_{z^{-1}} = \frac{1}{b_z} \tag{5.31}$$

$$b_{z^{-1}} = \frac{1}{a_z} \tag{5.32}$$

$$\mu_{z^{-1}} = a_{z^{-1}} + \left(\frac{m_z + n_z - 1}{m_z - 1}\right)(b_{z^{-1}} - a_{z^{-1}}); \text{from equation (5.16)} \tag{5.33}$$

$$\sigma_{z^{-1}} = \left(\left(\frac{1}{\|m_z - 1\|}\right)\sqrt{\frac{n_z(m_z + n_z - 1)}{(m_z - 2)}}\right)(b_{z^{-1}} - a_{z^{-1}}); \text{from equation (5.20)} \tag{5.34}$$

$$\{m_{z^{-1}}, n_{z^{-1}}\} \text{ calculated from: } \{\mu_{z^{-1}}, \sigma_{z^{-1}}\} \text{ and } \{a_{z^{-1}}, b_{z^{-1}}\} \tag{5.35}$$

using equations (5.3) and (5.5).

The structures equations (5.21) and (5.29) when recognized then:

$$Z \times Z^{-1} : \{1, \{1, 1\}, \{1, 0\}, \{\phi, \phi\}\}; \quad \phi\text{: signifies the empty set} \tag{5.36}$$

$$c : \{c, \{c, c\}, \{c, 0\}, \{\phi, \phi\}\}; \quad : \text{signifies the number } c \tag{5.37}$$

5.1.12 Special rule for Division

There is no explicit formula for division, the inverse object is used:

$$C = \frac{A}{B} = A \times B^{-1} \tag{5.38}$$

$$\text{where } C : \{c, [a_c, b_c], (\mu_c, \sigma_c), \{m_c, n_c\}\} \tag{5.39}$$

is obtained from equations (5.21) and (5.29).

5.1.13 Four ruls of Arithmetic

Addition and multiplication, with operators + and ×, respectively, are well defined. The other two operations, - and ÷, are defined via additive and multiplicative inverses:

$$X - Y = X + (-Y) \tag{5.40}$$

$$X \div Y = X \times (Y^{-1}) \tag{5.41}$$

equations (5.24) through (5.26) and equations (5.29) through (5.35) are respectively used in equations (5.40) and (5.41).

6 An Example to construct a Threat Contour Plot

6.1 Stages of LiveDesign Computational Engine Operations

Calibration and execution are carried out with simulated data and on-line real-time sensor and monitor data.

6.1.1 Calibration and 'training' the system

Following the schematic in Figure 6.1 a special case is illustrated in Figure 1.3. Instead of real-time on-line sensor data random data stream is generated for a wide variety of uncertainty. Ranges suitable for the Gaussian distribution, equation (**??**) to those applicable to extreme value distributions equation (**??**), are covered. Different degrees of fuzzy logic membership, small, for example 0.1 and below, to large .8 and above, are used for Gaussian and extreme value distributions, respectively, using the patterns yielded by Figure **??**:

$$\text{for example, 0.1 and below membership} \quad \text{:safety} \tag{6.1}$$

$$\text{for example, 0.8 and above membership} \quad \text{:security} \tag{6.2}$$

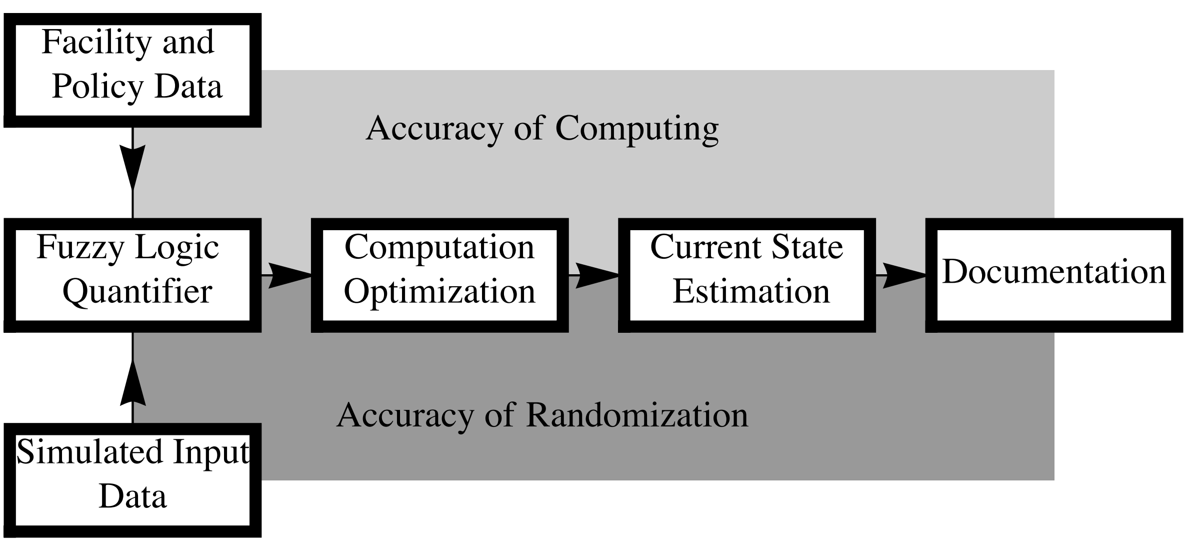


Figure 6.1: Verification of Accuracy in Randomization and Computing

There are several tests for the system before it is put to use. In a stage of calibration both the safety and security issues, Figure ??, are examined. Keeping the engineering plan and all other specifications constant, only different randomly generated sensor type data are used. This exercise is effective in fixing the bugs related to system failure described in Figure ?? and §??.

By varying engineering specifications mostly the falls alarm issues are addressed, and the programming bugs are fixed accordingly as described in Figure ?? and §??.

6.2 An Example of a collection of six rooms

The floor plan shown in Figure 4.12 is selected to represent a small clinic as shown in Figure 6.2.

Out[362]=

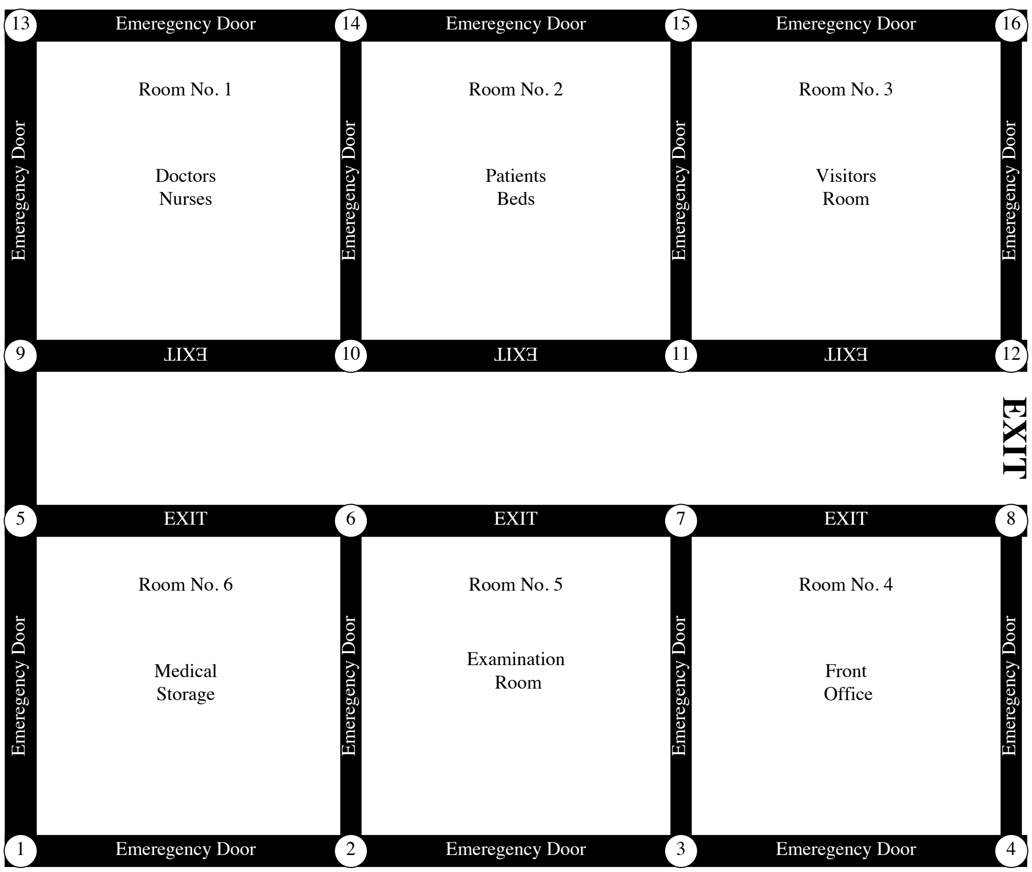


Figure 6.2: An Example of a Small Clinic

Intelligent sensor units are placed on the ceiling at the corners marked 1 though 16.

The sensor numbers are identified and their ranges are set according to their locations. All sensors my monitor events through out the height (or depth) of respective rooms. For example, in the horizontal plane inside the building, sensors at locations:

(i) 1, 4, 16 and 13 monitor 90 degrees

(ii) 2, 3, 15, 14, 5, 8, 12, and 9 monitor 180 degrees

(iii) 6, 7, 10 and 11 monitor 360 degrees

There can be 16 different sensor systems dictated by the human and material access. Different sensors perform different function, for example:

(i) sensor at location 1: detect medical and organic substances

(ii) sensors at location 8 and 12 count people form video inputs

(iii) sensor at location 16: monitor temperature and humidity for visitors comfort

At each location, such as, 1, 2, many different types of sensors can be combined to form a sensor cluster. The measurement of Mean Oxygen Flow at several locations may be the task of many chemical detecting sensors. Figure 6.3 shows (in the interest of clarity in the graphics, an exaggerated view of) the high and low values because it is assumed that there is a tolerance and error in any measurement. Figures similar to Figure 6.3 may be obtained for a number of pre-determined variables. In this example, only three measurements are demonstrated for brevity and clarity, without any loss in generality.

At different time the same variable may change in each room. For example, the temperature history for a predetermined short period of time can be used in numerical computation to obtain the mean, temperature, in Figure 6.4, and the standard deviation, in Figure 6.5, of the temperature in each room. The detail time histories of a generic quantity are shown in Figure 6.6, Figure 6.7 and Figure 6.8. The measurements are at stages 1, 2 and 3, those shown at intermediate stages, for example, 1.5 and 2.5, are the interpolated values.

Subsequently, all such data, like the temperature time history, mean oxygen flow, concentration of offending chemicals on the entire building, are combined to produce the threat contours, sown in Figure 4.13 and Figure 4.14.

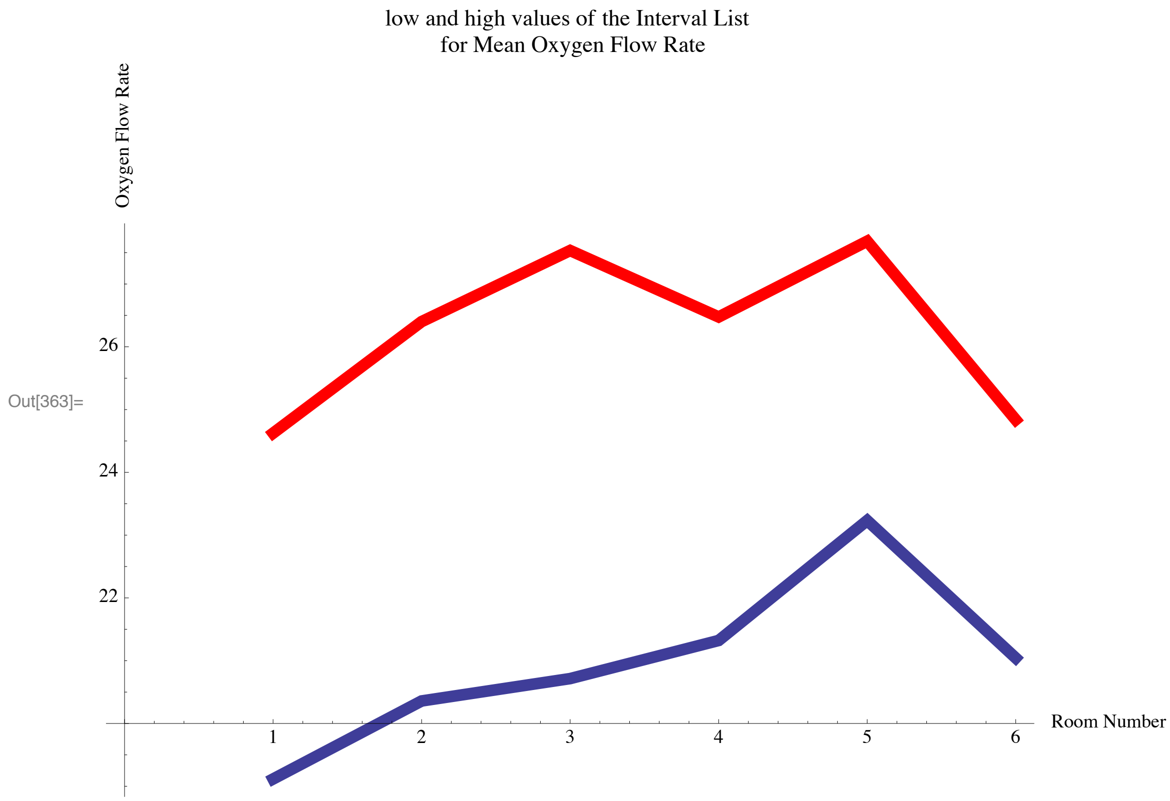


Figure 6.3: At a certain time, room-wise distribution of Mean Oxygen Flow

For this example, there are the following three variables at any time:

$$\text{Temperature}: \quad T \tag{6.3}$$

$$\text{Mean Oxygen Flow}: \quad f \tag{6.4}$$

$$\text{Offending Chemical}: \quad c \tag{6.5}$$

$$\text{and the threat value}: \quad \varphi = \varphi(T, f, c) \tag{6.6}$$

$$\text{and the threat value}: \quad \varphi \text{ for example, be Suffocation, Fire or Stampede}$$

The threat criteria, which has been selected by the emergency management, yields a value threat value φ as a function of T, f, c. In order to compute φ from equation (6.6) he four rules of arithmetic, §5.1.13, need to be performed when T, f, c are in the LiveDesignArithmetic form of equation (5.23). Using the calculation steps presented in §5.1.8,

$\varphi : \{\varphi, [a_\varphi, b_\varphi], (\mu_\varphi, \sigma_\varphi), \{m_\varphi, n_\varphi\}\}$

is obtained.

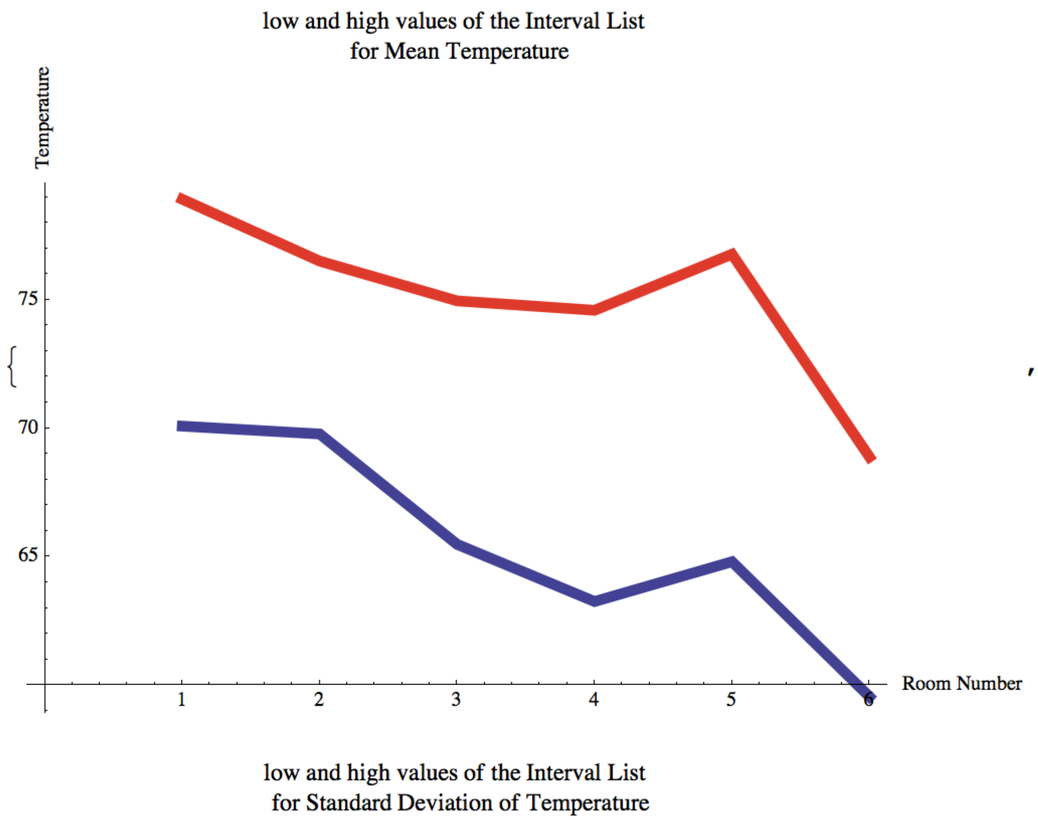
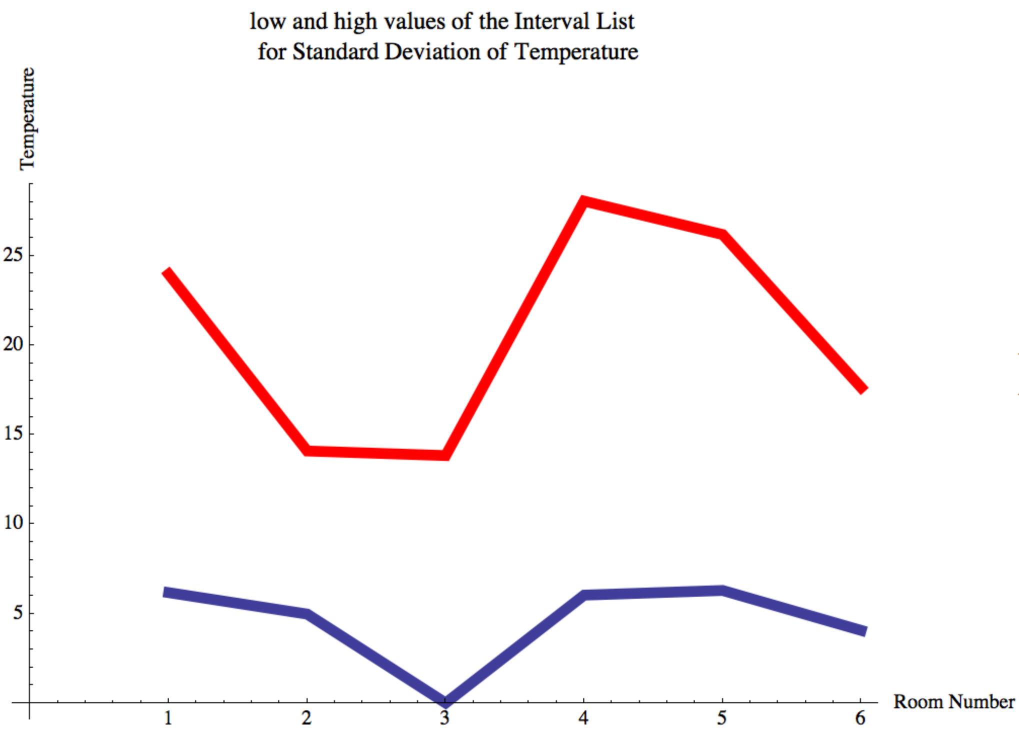


Figure 6.4: Over a certain period of time, room-wise distribution of Average Temperature



}

Figure 6.5: Over a certain period of time, room-wise distribution of the Standard Deviation of Temperature

In order to construct the threat contours a choice like the use of colors in the homeland security classification of threats, Figure 3.1, is to be used. Suppose there are four levels, none, low, medium and high, as shown in Figure 6.9.

Experts will seldom agree on a precise (crisp in the language of fuzzy logic) number for φ of equation (6.6). However, the experts may agree on the threshoulds defined by the lower and upper bounds. An example is shown in Figure 6.9. Now four colors, like Green, Blue, Orange and Red, may be used to plot the contours. Using fuzzy logic constructs some intermediate values, sometimes called the 'gray region ,' can be used. Such a threat plot at different time instants are shown in Figure **??** and Figure **??**.

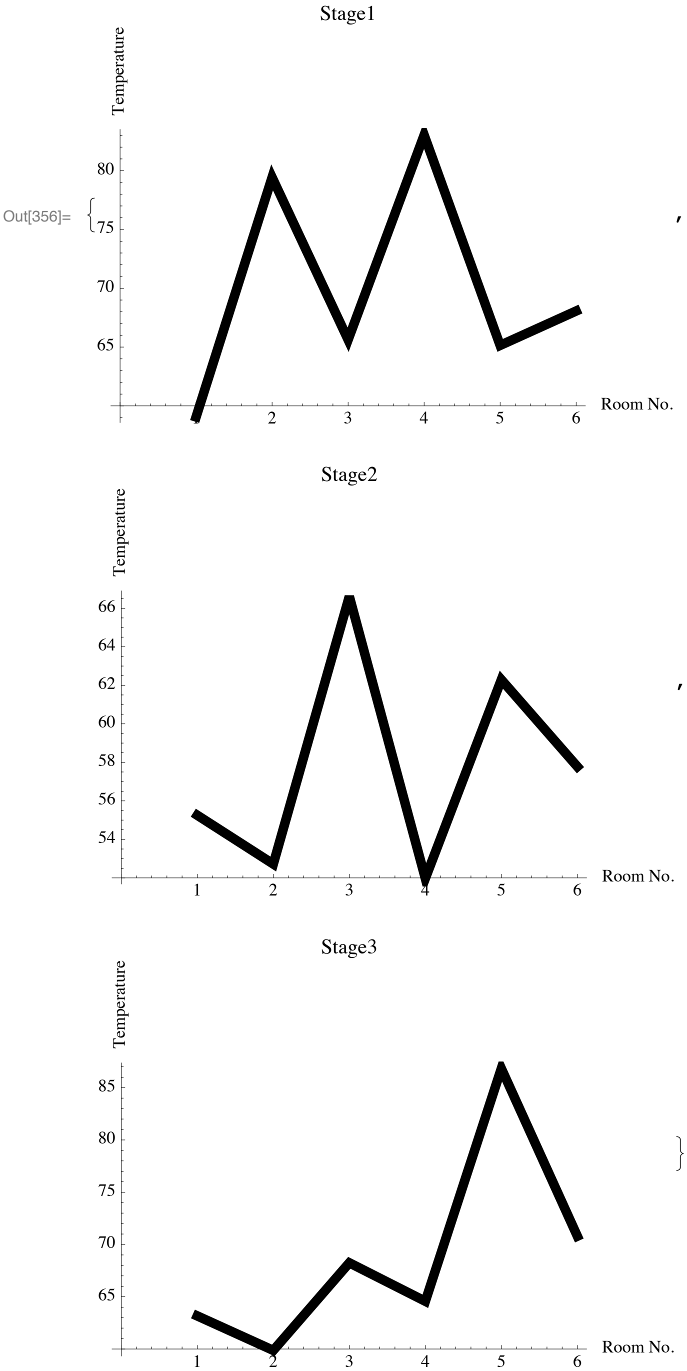


Figure 6.6: Temperature in different rooms in 3 time steps

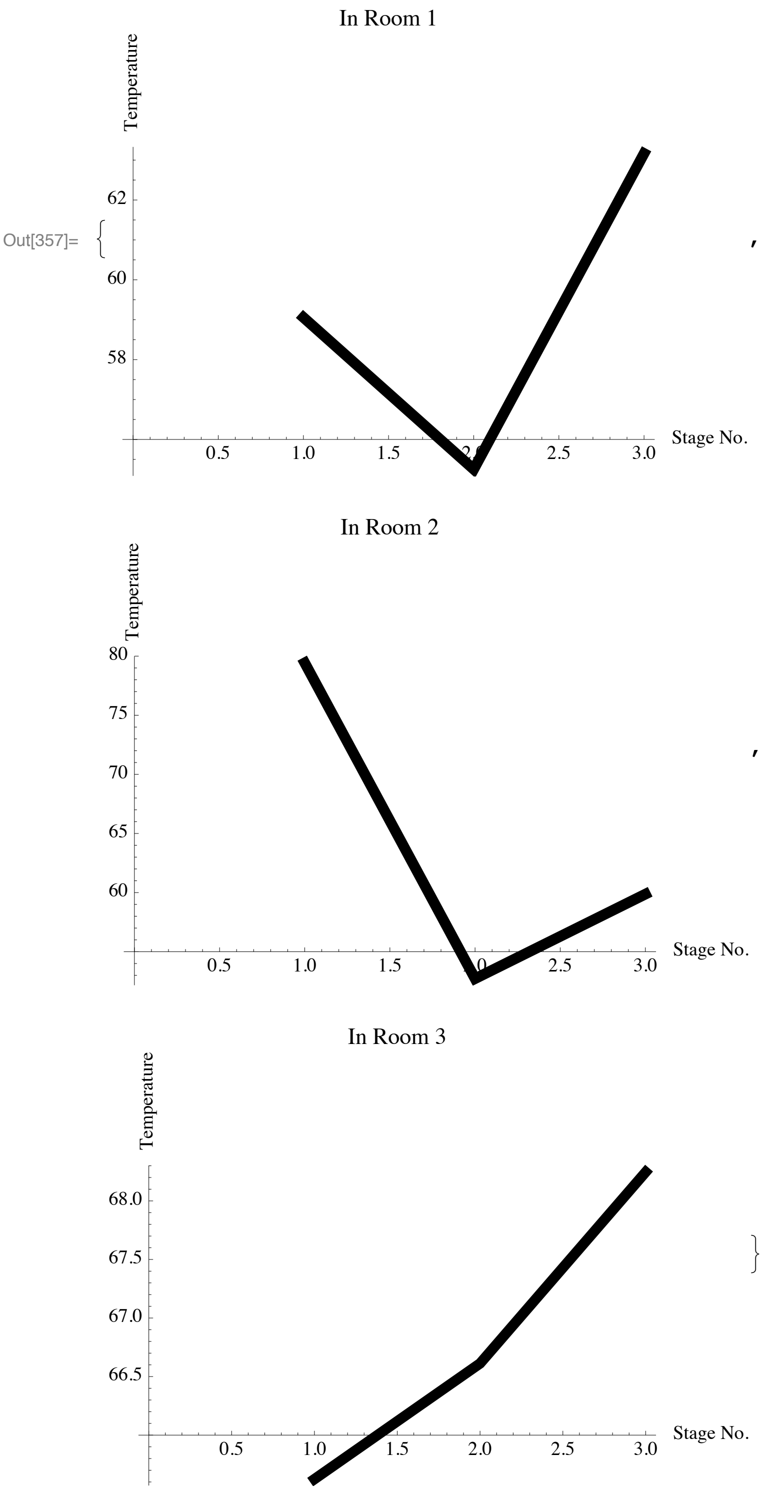


Figure 6.7: Temperature in different rooms in 3 time steps

Out[359]=

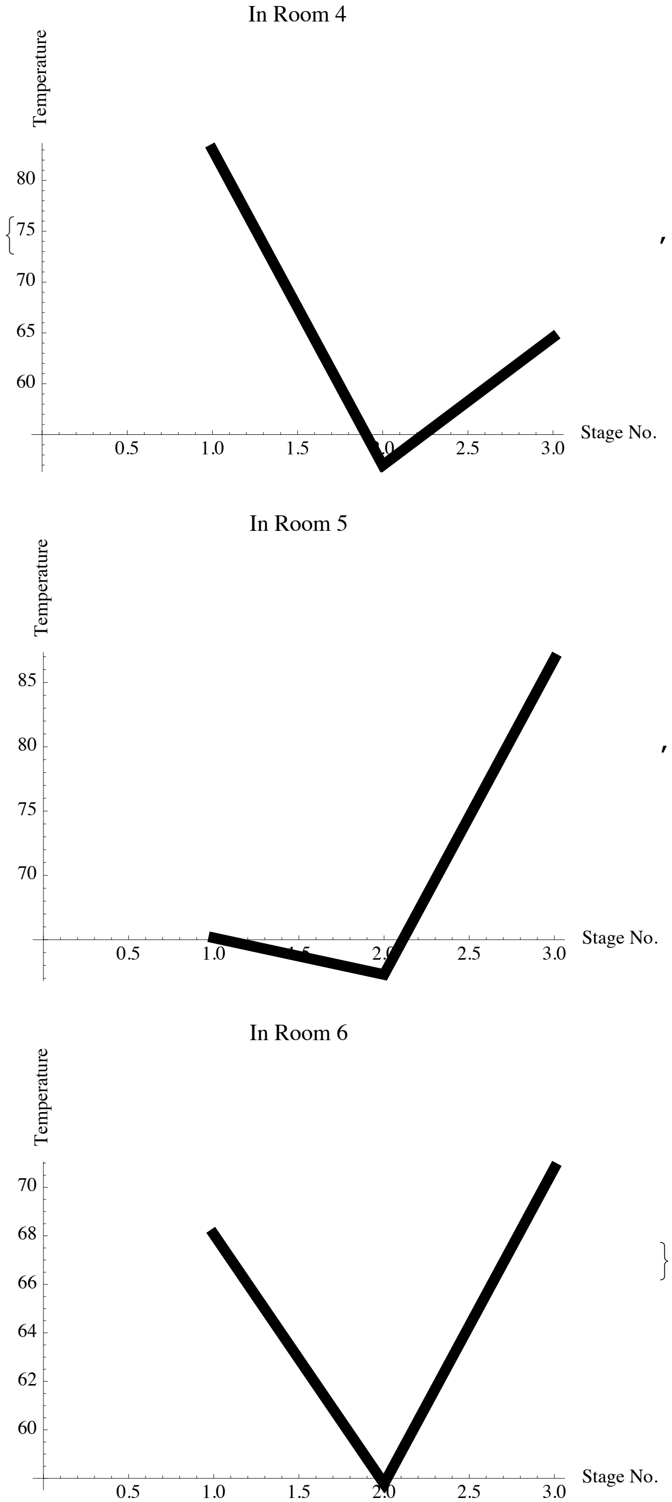


Figure 6.8: Temperature in different rooms in 3 time steps

	none	low	medium	high
Suffocation	(0,79)	(80,82)	(83,87)	(88,100)
Fire	(0,28)	(29,54)	(55,87)	(88,100)
Stampede	(0,3)	(4,8)	(9,44)	(45,100)

Figure 6.9: Fuzzy Logic-based threat clasification

List of Figures