

Development of an Expert System for Dust Explosion Risk Management Based on ASP.Net and Prolog

Qian Zhang

School of Materials and Metallurgy Engineering, Northeastern University, Shenyang, China
Email: zhangqian8647@163.com

Shengjun Zhong and Guanyu Jiang

School of Materials and Metallurgy Engineering, Northeastern University, Shenyang, China
Email: { zhongsj@smm.neu.edu.cn , heroius@163.com }

Abstract—Dust explosions always claim lives and cause huge financial losses. Dust explosion risk can be prevented by inherently safer design or mitigated by engineering protective system. Design of explosion prevention and protection needs comprehensive knowledge and data on the process, workshop, equipment, and combustible materials. The knowledge includes standards, expertise of experts, and practical experience. The database includes accidents, dust explosion characteristics, inherently safer design methods, and protective design methods. Integration of such a comprehensive knowledge system is very helpful. The developed system has the following functions: risk assessment, accident analysis, recommendation of prevention and protection solution, and computer aided design of explosion protection. The software was based on Browser/Server architecture and was developed using mixed programming of ASP.Net and Prolog. The developed expert system can be an assistant to explosion design engineers and safety engineers of combustible dust handling plants.

Index Terms—dust explosion, expert system, explosion prevention, explosion protection, ASP.Net, Prolog

I. INTRODUCTION

Dust explosions occur in a variety of industries and have a recorded history stretching back over 200 years. They always lead to some serious financial losses in terms of facilities damage and down time. They also often cause serious injuries to personnel, and fatalities. However, it is difficult for every equipment operator and technician to master the comprehensive professional knowledge and apply the knowledge to analyze and deal with the hazards because of the complexity of dust explosion affecting factors and the prevention and protection measures. Therefore, development of a dust explosion prevention and protection expert system using

existed expert knowledge, experience and problem-solving strategies instead of expertise of limited experts will reduce the hazards and losses dramatically.

The early attempts to develop expert systems (software) for dust-explosion vent design included Dust-Expert[1,2], ExTra[3]. Hesener and Kraus et al. have developed an expert system to be used to identify hazards due to the possible occurrence of various types of electrostatic discharges in various process situations. The system, using the CEN-ELEC report R044-001 as its technical basis, covers explosive gases/vapours/mists as well as explosible dusts[4].

Lorenz [5] has presented an expert system for the design of explosion venting arrangement based on the VDI 3673 venting code, which is very close to the new European Union code EN 14491[6]. The system accounts for the inertia of vent covers and doors and assesses forces acting on these covers and doors. The extent to which debris is ejected into the surroundings by destructive explosions is also accounted for.

Expert systems aimed to necessary auxiliary means of risk analysis of complex or disastrous fire hazard, such as risk assessment and risk management, were developed in the past 4 decades. In the 1970s, the U.S. National Bureau of Standards Center for Fire Research and Public Health Service Bureau developed a fire safety evaluation system that provided a unified approach to Fire Safety Assessment. Weng and Liao designed the basic fire risk assessment expert system based on semi-quantitative analysis[7]. Cai developed a fire and explosion risk assessment expert system using C-language[8]. The methods used included Dow Chemical Company (DOW) fire and explosion hazard index evaluation method, Richmond fire/explosion/toxicity evaluation indicators, Ammunition Enterprise quantitative assessment of major accident hazards et al. Qiu and Wang et al. designed a fire and explosion index safety assessment software using Visual Basic, which was mainly used in safety assessment of process plant in chemical companies[9]. There were many commercial risk assessment software available such as SAFETI, PHAST, FRED and SCOPE,

Manuscript received April 26, 2011; revised May 26, 2011; accepted May 30, 2011.

which can be used for impact reduction measures. These tools using quantitative risk assessment method can be applied once the process and instrumentation diagram of the design was completed[10].

The expert systems mentioned above are desktop applications. With the development and application of information technology, the traditional expert systems expose some problems and limitations[11]. Along with development of web technologies, it becomes a trend to build expert system on the internet, and it offers higher degree of sharing and expands the use widely that any internet user can log in the site running the expert system and consult through web browser. Because the program runs at the server and no custom application is delivered, the management and maintenance are much easier.

An expert system for dust explosion risk management named "DustEx.Net" was developed based on Browser/Server architecture. The theoretical foundation, architecture and key technologies were introduced.

II. FRAMEWORK OF DUST EXPLOSION RISK MANAGEMENT

The framework of dust explosion risk management is shown as Fig. 1. Risk assessment shall be carried out before any method of explosion prevention and mitigation is applied. Risk assessment includes risk analysis and risk evaluation. The most important parts of risk treatment are explosion prevention and explosion mitigation.

A. Methods of Dust Explosion Prevention and Mitigation

Methods of dust explosion prevention and mitigation are as follows:

- (1) Inherent safe design of process and equipment to prevent occurrences of ignition sources and dust cloud.
- (2) Apply inert atmosphere (inerting, or oxygen reduction).

- (3) Additional engineering protective methods
 - (a) Explosion venting;
 - (b) Explosion suppression;
 - (c) Explosion isolation;
 - (d) Containment (explosion resistant design).

Different methods have their own application conditions and limits, and the financial investments of different methods are of variety.

B. Dust Explosion Characteristics

Dust explosion characteristics (explosibility) are the input information of explosion risk assessment, and they are also used as supporting data in the design of explosion prevention and mitigation[12]. Explosion characteristics values for many materials can be found in the literature (e.g., NFPA 68[13] and Eckhoff[12]) or online databases. Such values should, however, only be used as references, while design of actual safety measures in industry should be based on test data for the actual dust in question [12]. The application of dust explosion characteristics is shown as Fig. 2.

The characteristics include:

- (1) Minimum ignition energy (MIE);
- (2) Minimum ignition temperature of dust layer (MIT-L);
- (3) Minimum ignition temperature of dust cloud (MIT-C);
- (4) Lower explosion limit (LEL), also called as minimum explosible concentration (MEC);
- (5) Limiting oxygen concentration (LOC);
- (6) Maximum explosion pressure p_{max} ;
- (7) Normalized rate of explosion pressure rise (explosion index) K_{St} ;
- (8) Dust volume resistivity (DR).

In risk assessment, MIE, MIT-L, MIT-C and LEL are characteristics to reflect the ignition likelihood, and p_{max} and K_{St} are characteristics to reflect the explosion severity (consequences). Likelihood has meanings of both probability and possibility.

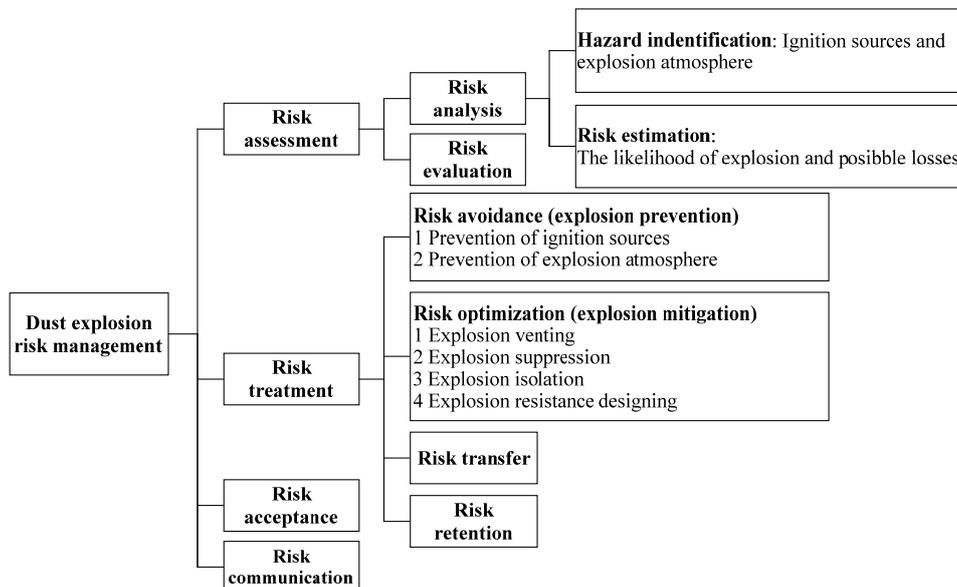


Figure 1. Framework of dust explosion risk management

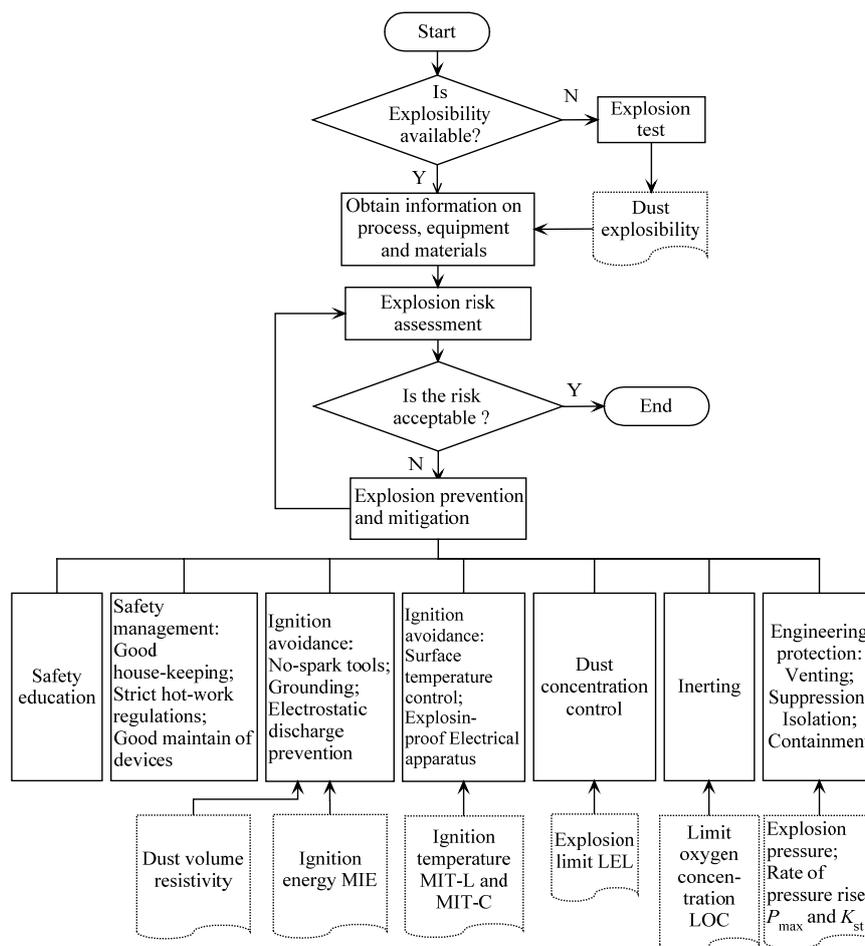


Figure 2. Flowchart of dust explosion risk management

III. ARCHITECTURE OF THE EXPERT SYSTEM

The name of the expert system is DustEx.Net, which means the system is based on Microsoft .Net Framework and B/S architecture. Currently DustEx.Net consists of five modules: accident analysis, risks assessment, prevention and mitigation decision making, explosion venting design and pressure-resistant design. Other modules might be added later.

The architecture of the expert system is shown as Fig. 3. The expert system includes databases, knowledge bases, inference engine, calculation models and a web server interface.

The database contains of historical accidents, dust explosion characteristics, typical processes, typical equipment, explosion prevention methods, explosion mitigation methods, protective devices, standards and references. The historical accidents database contains hundreds of the happened dust explosion hazard data, providing date, location, numbers of fatality and injury, description of the incidents, etc.. The dust explosion characteristics database contains published dust properties data on more than 1200 dusts, which should only be used as references. Users data can be added to the database. The database platform is Microsoft SQL Server. The relation schema of data sheets is shown as Fig. 4.

The knowledge bases contains: (a) the rules for decision making of explosion prevention and mitigation; (b) accident trees of typical processes and typical equipment. The knowledge bases are also called logic-bases, which are expressed as predicates in Prolog[14], and they can be compiled as dynamic link libraries (DLLs). Facts stored in database can be loaded to the knowledge bases dynamically by ODBC driver.

The calculation models are based on related international standards. The venting calculation is based on NFPA 68[13] and EN 14491[6]. Accurate sizing of vents is the most important aspect of vent design. The size of the vent depends on the explosion characteristics of the dust, the state of the dust cloud (concentration, turbulence and distribution), the geometry of the enclosure and the design of the venting device. The explosion characteristics of the dust used in vent sizing are the maximum overpressure P_{max} and the dust explosion constant K_{St} . The volume of the enclosure and the length-to-diameter ratio L/D relevant to the shape of the enclosure and the position of the explosion vent are required for vent sizing. The two parameters of vent device are the static activation overpressure P_{stat} and the weight per unit area of the venting element. The maximum value of the tolerance range of the static activation overpressure shall be considered. The weight

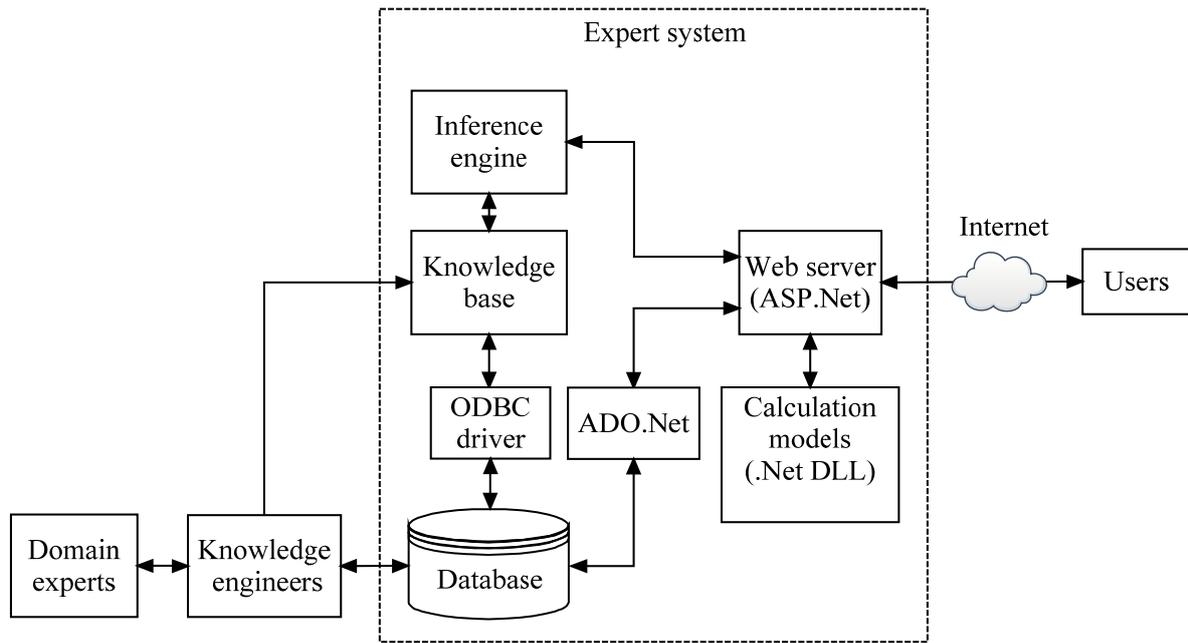


Figure 3. Architecture of DustEx.Net

per unit area of the venting element determines its venting effectiveness factor. Fig. 5 shows the core equations to calculate the vent area of vessels, silos and equipment without the effects of vent ducts and panel inertia according to NFPA 68 and EN 14491. In cases where rectangular enclosures like the basement of a silo, corridors or staircases need explosion pressure venting, the required vent areas can be calculated in another method. The general flowchart given in Fig. 6 shows the applicable selection of dust explosion vent sizing

calculation methods. In addition to the calculation of vent size, this module also allows the calculation of the reaction forces produced by a vented dust explosion and the extent of a fireball external to the vent.

The pressure resistant design is based on EN 14460[15] and ASME Section VIII, Division 1. The two principal vessel design parameters are the nominal thickness and allowable stress. The required thickness for the internal and external pressures will give an indication appearing

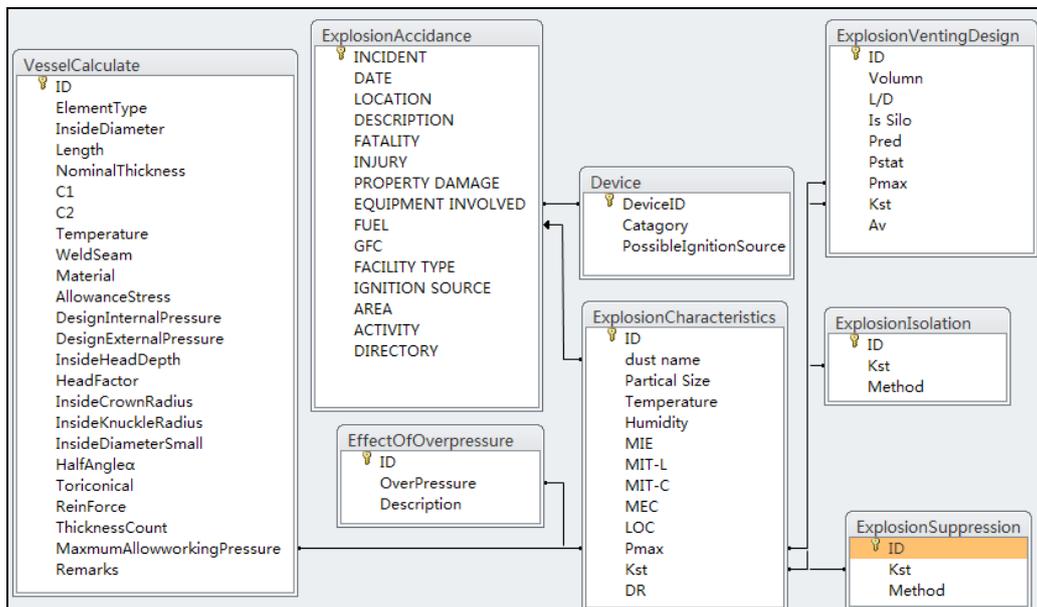


Figure 4. Relations among data tables

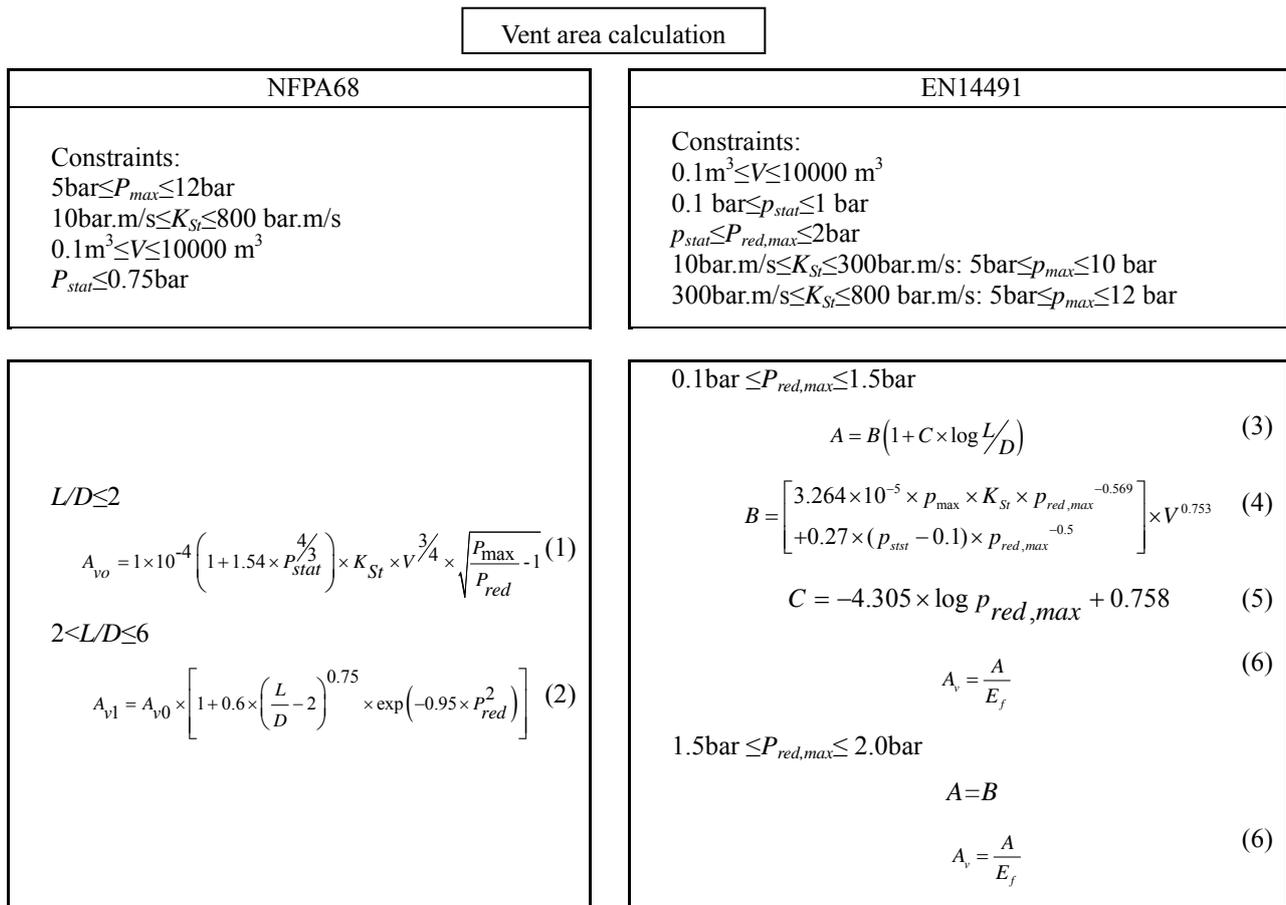


Figure 5. Core equations to calculate the vent area

Remark 1. P_{stat} -static activation pressure, bar; P_{red} -reduced pressure, bar; $P_{red,max}$ -maximum reduced explosion overpressure, bar; A_v -required vent area; V -volume,m³; L/D - length-to-diameter ratio.

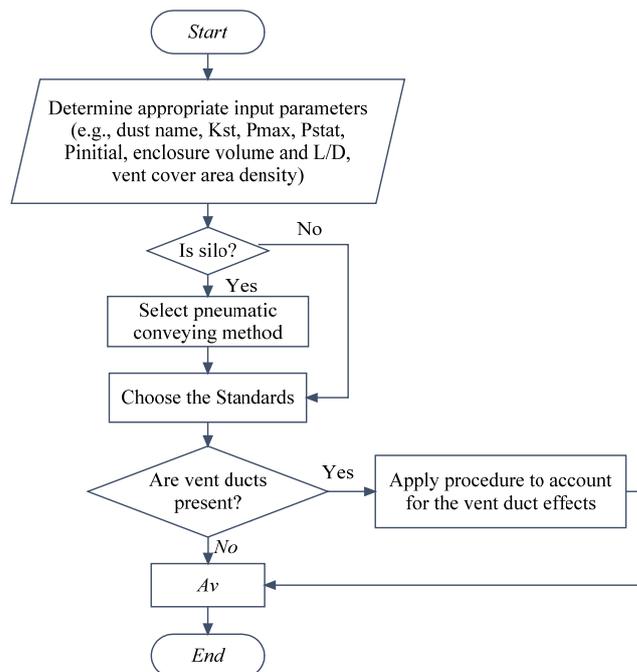


Figure 6. Flowchart for dust explosion vent sizing calculation

in red, if the nominal thickness is not adequate enough. The allowable stress will be calculated automatically after the material, the nominal thickness, the temperature and the design pressure. When the design temperature, or the nominal thickness of the shell or the heads is changed, the value of the allowable stress will be updated will automatically. Computational formulas of minimum required thickness and maximum allowable working pressure of typical element type (e.g., cylindrical shell, spherical shells, ellipsoidal head, torispherical head, conical head or shell segment) according to ASME Section VIII, Division 1 are shown in equations (7)-(18). The flowchart of the pressure resistant design is shown as Fig. 7.

The symbols defined below are used in the formulas:

t - minimum required thickness of the shells and heads, mm;

t_e - effective thickness, mm;

P - design internal pressure, MPa;

$[P_w]$ - maximum allowable working pressure, MPa

R - internal radius of shell course under consideration, mm;

D_i - inside diameter of the head skirt; or inside length of the major axis of an ellipsoidal head; or inside diameter of a cone head at the point under consideration measured perpendicular to the longitudinal axis, mm;

L - inside spherical or crown radius for torispherical and hemispherical heads, mm;

S - material allowable stress under design temperature, MPa;

E - weld seam factor;

K - a factor in the formulas for ellipsoidal heads depending on the head proportion $D_i/2h$;

M - a factor in the formulas for torispherical heads depending on the head proportion Ri/r ;

r - inside knuckle radius;

α - half the apex angle in conical heads and sections, deg.

Cylindrical shells:

$$t = \frac{PR}{SE - 0.6P} \quad (7)$$

$$[P_w] = \frac{SEt_e}{R + 0.6t_e} \quad (8)$$

Spherical shells:

$$t = \frac{PR}{2SE - 0.2P} \quad (9)$$

$$[P_w] = \frac{2SEt_e}{R + 0.2t_e} \quad (10)$$

Ellipsoidal heads:

$$t = \frac{PD_i K}{2SE\phi - 0.2P} \quad (11)$$

$$[P_w] = \frac{2SEt_e}{KD + 0.2t_e} \quad (12)$$

where

$$K = \frac{1}{6} \left[2 + \left(\frac{D_i}{2h} \right)^2 \right] \quad (13)$$

Torispherical heads:

$$t = \frac{PLM}{2SE - 0.2P} \quad (14)$$

$$[P_w] = \frac{2SEt_e}{ML + 0.2t_e} \quad (15)$$

where

$$M = \frac{1}{4} \left(3 + \sqrt{\frac{L}{r}} \right) \quad (16)$$

Conical heads or shell segment:

$$t = \frac{PDi}{2 \cos \alpha (SE - 0.6P)} \quad (17)$$

$$[P_w] = \frac{2SEt_e \cos \alpha}{D_i + 1.2t_e \cos \alpha} \quad (18)$$

Fuzzy fault tree analysis method [17-20] is used in the risk assessment models. Fuzzy logic deals with uncertainty and imprecision. It is an efficient tool for solving problems when there are no clear boundaries between failure and success states of the systems, when the probability of the system failure cannot be calculated precisely due to lack of sufficient data, and when there is subjective evaluation of the reliability, which is made with natural language expressions. Such situations frequently arise in a quantitative fault tree analysis in safety and risk assessment of different processes. In the classic fault tree analyses, the input variables are treated as exact values and the exact outcome data are received by an appropriate mathematical approach. In the fuzzy method, all variables are replaced by fuzzy numbers in the process of fuzzification and subsequently using fuzzy arithmetic, fuzzy probability of the top event for fault tree. The calculation models were written in C#, and were compiled to be .Net DLLs.

An inference engine named ExProlog was developed using C#, and it will be introduced in section IV.

The web server was developed in ASP.Net platform using C#, JavaScript and HTML. It serves as the user interface and it calls: (a) the inference engine for logical inference; (b) ADO.Net for database management; and (c) calculation models for risk assessment, venting design and explosion resistant design.

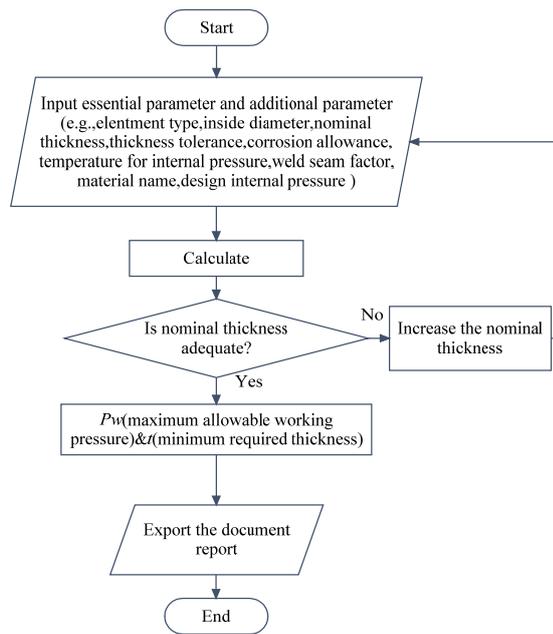


Figure 7. The flowchart of the pressure resistant design

IV. KEY TECHNOLOGIES

A. Knowledge Base Construction using ExProlog

Prolog is the most popular logic programming language used to develop expert systems. It is a declarative language, which is quite different with procedural languages such as C and Java. It is powerful in logical inference such as solution searching and reason tracing.

The developed component ExProlog provides an inference engine running on .NET framework, which follows the characteristics of Prolog to execute clauses non-sequentially, and its new attribute is that the rules are definitely independent from the code, which means new knowledge could be added dynamically into the runtime knowledge base.

As basic support, C# classes of *Clause*, *Premise* and *Domain* are defined corresponding to their traditional concepts. An instance of *Clause* could be either a fact or a rule that is used to form the knowledge base; an instance of *Premise* presents an uncertain state which can be confirmed through the solving function of inference engine; an instance of *Domain* is essentially a wrapped variable included in the parameter list of a *Clause* or *Premise* instance.

Engine class was designed as an environment containing the runtime knowledge base and the inference engine. Users may change some Boolean properties to control the functions of *Engine* instances, like telling the engine if there is a need to reason or to record the path when calls the *Goal* method. Some datasets were set in the class, respectively applied for saving initial facts (explicitly added), rules, and expanded facts (initial facts and the results from reasoning) - the first two kind of knowledge could be imported from external by users. Serial of methods were supplied around the knowledge base consisting of the datasets, and the most basic one of

them is the *Goal* method, which implemented the solving function. This method accepts a *Premise* instance, tries to find the matched clauses in the knowledge base using predicate as the heuristic condition, and if the premise is hit, returns the solutions in a list of domain array. Fig.8 shows the flowchart of the *Goal* method. Based on the *Goal* method, the *Reason* method, that may be called every time after a new fact is added to the knowledge base, implemented the reasoning function. The process of reasoning is to find rules, the sub-premises of which contains one that matches with the new fact - if such a rule is found, it will be abstracted into a relative premise, and the *Goal* method will be called with this premise, and the returned values are, after being transformed into facts

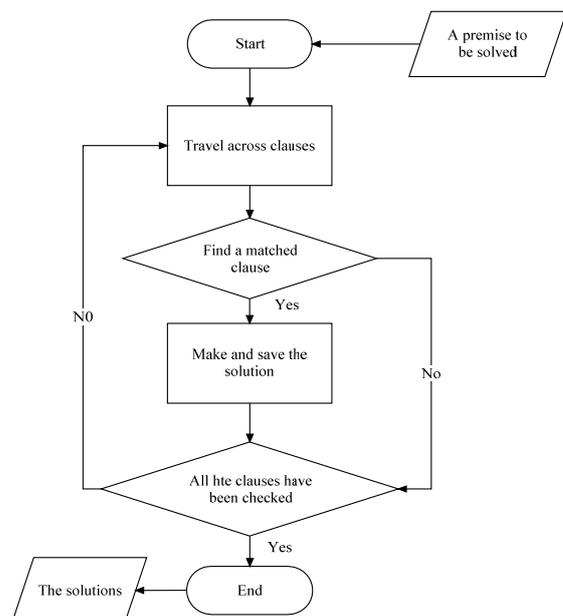


Figure 8. The flowchart of the Goal method

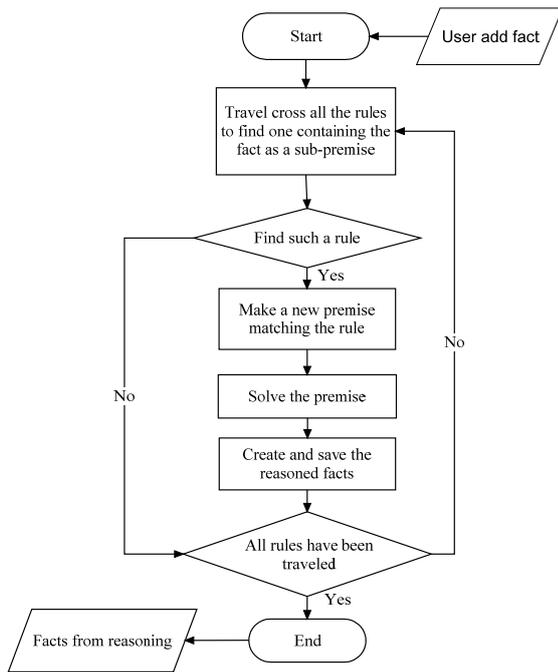


Figure 9. The flowchart of the Reason method

actually, the facts from reasoning, which would be added automatically into the expanded facts dataset. Fig.9 shows the flowchart of the Reason method. Based on the Reason method, the Query method implemented a similar function with the Goal method. The difference is the Query method using the expanded facts dataset to do solving job rather than the initial facts and rules datasets. This reduces the resource consumption when there is no need to record the solving path.

When building an expert system with ExProlog, we mainly use the reasoning function of the inference engine. As every time the user answers the question given by the system, a new fact based on the answer is created and added, and the Reason method returns the facts reasoned, according to which the system can tell what to ask next. The expert shell takes charge in inputting added fact to knowledge base and getting returned facts to decide whether it's time to stop or keep asking another question. The real expert knowledge is totally saved in data files, generally including a fact as the first asked question and several rules that response the user's answers.

Table 1 describes how to take protections depending on MIE [21]. In order to simplify discussion, the information in the table is considered as a set of typical

TABLE I. RECOMMENDED SAFETY ACTIONS DEPENDING ON MIE

Ignition sensitivity	MIE(mJ)	Safety actions
Normal	MIE>10	Innition source avoidance is adequate
Special	10≥MIE≥3	Expert decision required
Extreme	3>MIE	Additional to ignition source avoidance, explosion protective system are required

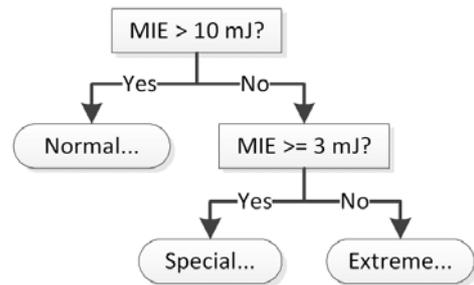


Figure 10. The logic diagram of expert system according to MIE expert knowledge. To build an according expert system, analyze each possible cases (true or false), take these cases as mid nodes of a tree and their results as leaf nodes, then the system structure will be like the one shown in Fig. 8.

So the first asked fact is:
 ask ("MIE>10?")
 and the responding rules are:
 ask ("MIE>=3?"):-answer("MIE>10?N").
 result ("Normal: Ignition source avoidance is adequate."):- answer("MIE>10?Y").
 result ("Special: Expert decision required."):- answer("MIE>=3?Y").
 result ("Extreme: Additional to ignition source avoidance, Explosion protective systems are required."):- answer("MIE>=3?N").

Those two types of clauses are saved in separated files and imported as the program starts. Once the system loads these clauses, it begins to ask questions by itself. Users may just answer Yes or No to the question and a new fact with the predicate “answer” based on the question and the answer is added to the knowledge base. The Engine instance calls the Reason method to deal with the added fact, and returns a reasoned fact to make the next question. If the returned fact's predicate is “result”, the system stops.

B. Mixing Programming of ASP.Net and Prolog

By using Prolog, the programmer only needs to tell the computer facts and rules, and solution search is the business of Prolog's reference engine. However, Prolog is not suitable for mathematical calculation or user interface development.

Microsoft .Net Framework is a new generation of development platform for both desktop applications and web applications. ASP.Net is a web development environment for many different languages such as C#, VB.Net, F# and Delphi Prism. Traditional Javascript and HTML script can also be used in ASP.Net. So mixing programming of Prolog and ASP.Net can take the advantages of Prolog and ASP.Net and avoid the disadvantages. Many commercial Prolog compilers provide programming interface for C. C# can use the same interface by COM invoking.

V. CONCLUSIONS

A framework of expert system for dust explosion risk management was constructed. The functions include risk analysis, accident cause analysis, explosion prevention

and mitigation decision making, venting design and explosion resistant design. The application is based on Brower/Server architecture, which enables users to access the developed system by Internet.

Prolog is good at solution searching and reason tracing, and it is a promising language for development of complex expert system. ASP.Net is a web development platform which provides powerful tools for development of user interface, database management, as well as mathematical calculation. Mixing language programming can take the advantages of both Prolog and ASP.Net.

The expert system is under construction, however, preliminary case tests proved that the finished modules worked very well.

REFERENCES

[1] Abbasi, T., Abbasi, S.A.: Dust Explosions—Cases, Causes, Consequences, and Control. *J. Journal of Hazardous Materials*. 140, 7--44 (2007).

[2] Vadera, S., Meziane F., Huang M.-L.L.: Experience with Mural in Formalizing Dust-Expert. *J. Software Technol.* 43, 231--240 (2001).

[3] Hesener, U., Schecker H.G.: ExTra—An Expert System for the Safety Analysis of Drying Plants. In: Mewis, J.J., Pasma, H.J., De Rademaeker, E.E. (eds.) *Loss Prevention and Safety Promotion in the Process Industries*. vol. 2, pp.643—653. Elsevier, Amsterdam (1995).

[4] Hesener, U., Kraus, A., Schecker, H.G.: Computer Based Assessment of Hazards Caused by Electrostatic Discharges. In: *Proceedings of the Ninth International Symposium on Loss Prevention and Safety Promotion in the Process Industry*, pp. 851—859. Barcelona, Spain (1998).

[5] Lorenz, D.: ExProtect: A Software Response to Important to Important Questions on Safety in Dust and Gas Explosion. In: *Proceedings of the Third World Seminar on the Explosion Phenomenon and on the Application of Explosion Protection Techniques in Practice*. Ghent, Belgium (1999).

[6] EN 14491. *Dust Explosion Venting Protective Systems* (2006)

[7] Weng, T., Liao, G.: Discussion on Systematic Construction of Expert Software for Assessment of Fire Risk Gradation. *J. China Safety Science Journal*.15, 37--40(2005).

[8] Cai, Z.: Evaluation of Fire and Explosion Hazard and Development of Expert System Software. *J. Mining Safety and Environmental Protection*.1--3(2004).

[9] Qiu, K., Wang, Q.A.: A Primary Software Design in Fire Explosion Index Safety Evaluation. *J. Chongqing Institute of Technology Journal*. 8, 102--105(2006).

[10] Shariff, A.M., Rusli, R., Leong, C.T., Radhakrishnan, V.R., Buang, A.: Inherent safety tool for explosion consequences study. *J. Journal of Loss Prevention in the Process Industries*. 19, 409--418(2006).

[11] Duan, Y., Edwards, J.S., Xu, M.X.: Web-Based Expert Systems: Benefits and Challenges, *Information and Management*.42 (2005).

[12] Eckhoff, R.K.: *Dust Explosions in the Process Industries*, 3rd ed. Gulf Professional Publishing/Elsevier, Boston (2003).

[13] NFPA 68. *Standard on Explosion Protection by Deflagration Venting* (2007).

[14] Bramer, M.: *Logic Programming with Prolog*. Springer (2005)

[15] EN 14460. *Explosion Resistant Equipment* (2007)

[16] ASME Code, Section VIII, Division 1. *Rules for Construction of Pressure Vessels* (2007).

[17] Dokas, I.M., Karras, D.A., Panagiotakopoulos, D.C.: Fault Tree Analysis and Fuzzy Expert Systems: Early Warning and Emergency Response of Landfill Operations. *J. Environmental Modelling and Software*.24, 8--25(2009).

[18] Markowski, A. S., Mannan, M. S., Bigoszevska, A.: Fuzzy Logic for Process Safety Analysis. *J. Journal of Loss Prevention in the Process Industries*. 22, 695--702(2009).

[19] Bajjaia, S., Sachdevab A., Gupta, J. P.: Security Risk Assessment: Applying the Concepts of Fuzzy Logic. *J. Journal of Hazardous Materials*. 173, 258--264(2010).

[20] Berta, I.: Use of Soft Computing Methods in Risk Assessment of Electrostatic Fire and Explosion Hazards in Industries. *J. Journal of Electrostatics*. 67, 235--241(2009).

[21] Pekalski, A. A., Zevenbergen, J. F., Lemkowitz, S. M., Pasma, H. J.: A Review of Explosion Prevention and Protection Systems Suitable as Ultimate Layer of Protection in Chemical Process Installations. *J. Process Safety and Environmental Protection*. 83(B1), 1--17(2005).



Qian Zhang(Anshan, Liaoning province, China, 04-07-1986). Master degree of environmental science, school of material and metallurgy, Northeastern University, Shenyang, China, 2011.

She worked in Industrial Explosion Protection Institute of Northeast University as a Postgraduate since 2009. She is interested in software development and explosion risk

assessment.



Shengjun Zhong (Hubei province, China, 01-13-1970). Ph.D of explosion dynamics, Institute of Heat Engineering, Warsaw University of Technology, Poland, 2003.

He worked in School of Materials and Metallurgy Engineering, Northeastern University as Associate Professor since 2002. He is interested in explosion safety, risk assessment and software

development.



Guanyu Jiang (Shenyang, Liaoning Province, China, 02-25-1986), bachelor of environment science, school of material and metallurgy, Northeastern University, Shenyang, China, 2009.

He worked in School of Materials and Metallurgy Engineering, Northeastern University as Science Assistant since 2009. He is interested in software development