Study of Evacuation in High-rise Buildings

And

Attempt of Evacuation Plan Devisal with Functional Knowledge

高层建筑安全疏散问题的研究
及泛函背景下疏散方案设计的尝试

Hangzhou Foreign Languages School

杭州外国语学校

Datong Zhou　　Yuhan Wang

周大桐　王雨菡

Guided by:　Jun Pan

指导教师　潘俊

August, 2013

Abstract

This article applies functional knowledge and computer software to describe the relation between people distribution and evacuation efficiency and safety in high-rise buildings. The people distribution contains two parts: the allocation of people in different storeys and the distribution of people in a certain storey.

Through this mathematical model, we attempt to find the optimum plan of people distribution and use it to guide the distribution of different functional areas in high-rise buildings.

Key Words: functional; people distribution; evacuation efficiency and safety; high-rise buildings

Contents

Chapter Ⅰ Introduction

From the 2008 Wenchuan Earthquake to the mud-rock flow in Zhouqu in 2010, natural disasters have repeatedly occurred in our country in recent years, including the Ya'an Earthquake this year which gave rise to heavy injuries and deaths. In order to better react to possible disasters, our school organized a mimic evacuation last autumn, and it was in this evacuation that an idea struck us to put forward a brand-new perspective in countermeasures against potential emergencies.

After some research, we have found that the academia puts major emphasis on the following four aspects to decrease casualties:

1. Ensuring the security of the buildings
2. Optimizing the layout of the structures
3. Designing the optimum evacuation route
4. Arranging rescue plans and medical treatment

However, we noticed another factor that might also exert a strong influence on the efficiency and security during the evacuation. That is, the distribution of people among different areas, including people allocation in different storeys and people distribution in a specific storey.

In reality, high-rise buildings are multi-functioned. There may be places of office work, commerce, accommodation, catering, etc. There will be different people densities according to the specific function of each storey. Therefore, we can modify the functions of each storey to regulate the people allocation. Take shopping malls, for instance, we can modify the distribution of clothing stores, restaurants and depositories to change the number of customers in each storey and their possible distribution in a certain storey. In this manner, the overall economic benefits can be retained while the evacuation will be more successful.

Confirming the feasibility of this method, we set out to construct a quantitative model from the simplest form of a three-storey building. Based on the premise that people from the first floor need not go by the stairway to evacuate, we attempt to come up with an equation that can describe the evacuation efficiency at every stairway exit. During the process, we need the relation between the average velocity and the people density. After consulting some data and documents, we find that this equation often appears in the form of negative exponent function. But we realize, after on-site measurement, that this equation is not quite consistent with our model. Thus, we optimize the original one and adopt a more reasonable curve to describe the relation between the average velocity and the people density in a certain stairway. Ultimately we obtain a relatively precise set of equations.

With the assistance of mathematical tools, we obtain, in quantitative terms, the influence of people distribution on the evacuation efficiency and are therefore able to find out the best people distribution method. Meanwhile, considering the possible impact of large number of rush-into people at the stairway exit, we describe it with the disturbance of each locale to observe and analyze its effect on the final outcome. Then we come up with the relation between the people distribution and the evacuation risk.

To conclude, we can modify, according to their significance, the weights of the two factors (evacuation efficiency and security) to obtain the best people distribution pattern in a manner that can meet the needs of both, so as to guide the inner layout of the building and further guarantee the life security of the people when disasters occur.

Chapter Ⅱ Related Knowledge

2.1 Partial Derivative

In mathematics, a **partial derivative** of a function of several variables is its derivative with respect to one of those variables, with the others held constant (as opposed to the total derivative, in which all variables are allowed to vary). The partial-derivative symbol is ∂.

2.2 Ordinary Differential Equation(ODE)

In mathematics, an **ordinary differential equation** or **ODE** is an equation containing a function of one independent variable and its derivatives.

2.3 Calculus of Variations

Calculus of variations is a field of mathematical analysis that deals with maximizing or minimizing functionals, which are mappings from a set of functions to the real numbers. Functionals are often expressed as definite integrals involving functions and their derivatives. The interest is in extremal functions that make the functional attain a maximum or minimum value – or stationary functions – those where the rate of change of the functional is zero.

2.4 Linear algebra

Linear algebra is the branch of mathematics concerning vector spaces, often finite or countably infinite dimensional, as well as linear mappings between such spaces. Such an investigation is initially motivated by a system of linear equations containing several unknowns. Such equations are naturally represented using the formalism of matrices and vectors.

Chapter Ⅲ Model Building

3.1 Function Description of evacuation efficiency in stairways

3.1.1 Process of Model Building

When a stairway is short enough, the lag effect caused by the rush-into crowd can be ignored, so we can consider that there is a certain function relation between the velocity of circulation and the people density. Obviously, this function must satisfy the following requirements:

1. As x increases on the positive semi axis, the function value must increase at first, and then decreases.

2.When x is 0, the function value must be 0.

3.When x is on the positive semi axis, the function value is always positive.

4.The function must always have value in real number domain for the convenience in our later calculation.

Generally, the academia studies on the circulation velocity and people density in a crowded condition by means of on-site observation and picture recording. Up to now, a large number of data has been accumulated. Among the experts studying on it, American scholars Fruin, Maclennan, Nelson, English scholar Smith, Japanese scholar Ando and Canadian scholar Paul are very representative. These experts have obtained quite distinct function relations between the circulation velocity and people density in a certain short stairway, such as linear functions, circular functions, logarithm functions and negative exponent functions. [1] Negative exponent functions are most widely used.

However, on the basis of our observation in our school's stairways, we find that when x tends to be very large, the function still has a small value. It's quite different from the negative exponent functions' prediction, which says the value will tend to approach 0 rapidly.

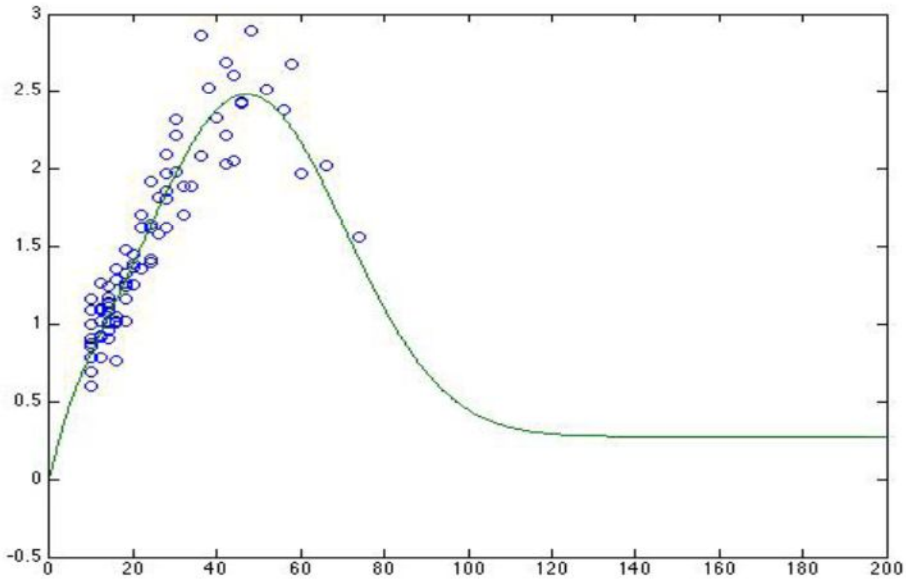
Thus, we attempt to adopt truncated normal transformation to fit this relation: suppose $s(x)$ is the target function,

$$s(x) = v_0 + v_1 e^{-k_1\left(\frac{x}{S}-\rho_1\right)^2} - \left(v_0 + v_1 e^{-k_1(\rho_1)^2}\right) e^{-\frac{k_2 x}{S}}$$

x is the independent variable unit, which means the aggregate number of people in a certain stairway. S is the area of the stairway. The third term is the penalty function, which can make the function value 0 when x is 0. Moreover, this term's existence ensures that the optimum solution will not go beyond the domain in reality application.

3.1.2 Practical Test

According to our on-site measure and calculation in our school's stairways, we have obtained enough data to fit the relation and draw the fitting curve, which is:



The fitting result is:
s=26.77（obtained by measure） ρ_1=1.760
k_1=0.6519 k_2=5.1020 v_0=0.2752 v_1=2.2079

3.1.3 Comparison with Other Results in Academia

On the basis of the form of our function, we can know:

$$v_{max} \approx v_0 + v_1 = 2.4831$$

$$\rho|_{v=v_{max}} \approx \rho_1 = 1.760$$

The effective circulation velocity at per unit of width is about 1.6020.

American scholar Fruin's result, which is obtained by statistics, is [2]:

$$\rho|_{v=v_{max}} \approx 2.0$$

The effective circulation velocity at per unit of width is about 1.18.

The national key fire science laboratory in University of Science and Technology of China obtains its result by the means of camera shooting, which is [3]：

$$\rho|_{v=v_{max}} \approx 1.8$$

3.2 Model in Weak Conditions

3.2.1 Process of Model building

Suppose that each building has a corresponding time and we call it safe time t_0. During this period of time after emergencies, we can ensure the building's safety. After the safe time, the building's safety cannot be guaranteed. Therefore, our purpose is to achieve the most successful evacuation in the safe time.

Firstly, we discuss in the conditions that we can only modify the allocation of people in different storeys, but can't modify people's distribution in a certain storey. So we temporarily assume that the relation between the time and the number of rush-into people in per unit of time can be described as negative exponent function (will be optimized in 3.3). Thus, to each stairway, we can use a corresponding equation to describe the variation of people's quantity in this stairway.

Generally, we suppose $s(x)$ is the evacuation velocity when the number of people in the stairway is x. And $y_i(t)$ is, as time goes by, the change of the quantity of rush-into people in per unit of time from the storey i. Thus, we can obtain this equation:

$$\frac{dx_i}{dt} = -s(x_i) + s(x_{i-1}) + y_i(t)$$

Regard evacuation efficiency as:

$$s(x) = v_0 + v_1 e^{-k_1\left(\frac{x}{S}-\rho_1\right)^2} - \left(v_0 + v_1 e^{-k_1(\rho_1)^2}\right)e^{-\frac{k_2 x}{S}}$$

Therefore, we can take the simplest three-storey building for instance and assume that people from the first floor can evacuate without going by stairways. Then we obtain this equation system:

$$\frac{dx_1}{dt} = f_1 = -s(x_1) + \alpha e^{-mt}$$

$$\frac{dx_2}{dt} = f_2 = -s(x_2) + s(x_1) + (A - \alpha)e^{-mt}$$

$$\frac{dx_3}{dt} = f_3 = s(x_2)$$

x_3 is the quantity of people who has already successfully evacuated. A is determined by aggregate number of people, while α is determined by allocation plan, so we need to modify α to find the optimum allocation plan.

x_i.

3.2.2 Mathematical Proof

To a first order ordinary differential equation system, which has n fixed initial-values and a fixed parameter α,

$$\frac{dx_i}{dt} = f_i(x_1, \dots, x_n, t, \alpha) \;\; i = 1,2 \dots n$$

$$x_i(0) = x_i|_{t=0}$$

Its image is a curve in $(n+1)$-dimensional space. Leading the $(n+2)$ dimension α into the system, we will get a certain curve surface corresponding to a certain value of α. All these curves make up a curved surface in $(n+2)$-dimensional space.

Considering α as a variable quantity and change all the derivatives to t into partial derivatives to t, we will therefore obtain a partial differential system. The boundary condition is that all initial-values should be known when t equals 0:

$$\frac{\partial x_i(t,\alpha)}{\partial t} = f_i(x_1 \dots x_n, t, \alpha) \;\; i = 1,2 \dots n$$

$$x_i(0,\alpha) = x_i|_{t=0}$$

To a certain equation system, the solution is unique.

We do the partial derivative to t, and obtain n equations:

$$\frac{\partial^2 x_i(t,\alpha)}{\partial t \partial \alpha} = f_i(x_1 \dots x_n, t, \alpha) \;\; i = 1,2 \dots n$$

As α can't influence initial-values, we can say:

$$\frac{\partial x_i(0,\alpha)}{\partial \alpha} \equiv 0$$

According to that, we can build up n auxiliary equations:

$$\frac{d}{dt}\frac{\partial x_i(t,\alpha)}{\partial \alpha} = f_i(x_1 \dots x_n, t, \alpha) \;\; i = 1,2 \dots n$$

$$\frac{\partial x_i(0,\alpha)}{\partial \alpha} = 0$$

$\frac{\partial x_i(0,\alpha)}{\partial \alpha}$ means, on the curved surface in $(n+2)$-dimensional space, the slope in the direction of α (partial derivative) at an arbitrary moment.

From the angle of function, there is a certain value corresponding to a certain (t,α). Thus, we can regard the system as a binary function to (t,α). Then the problem of seeking x_i's extreme value has been transformed into the problem of enabling $\frac{\partial x_i(t_x,\alpha)}{\partial \alpha} = 0$. The latter is a two-point boundary value problem, which we can easily solve and therefore obtain the optimum α and extreme value of x_i.

3.2.3 Application in This Model

According to basic properties of partial derivatives, we can know:

$$\frac{d\frac{\partial x_1}{\partial \alpha}}{dt}=\frac{\partial f_1}{\partial \alpha}$$

$$\frac{d\frac{\partial x_2}{\partial \alpha}}{dt}=\frac{\partial f_2}{\partial \alpha}$$

$$\frac{d\frac{\partial x_3}{\partial \alpha}}{dt}=\frac{\partial f_3}{\partial \alpha}$$

$$\forall\frac{\partial x_i}{\partial \alpha}(0)=0;\frac{\partial x_3}{\partial \alpha}(t_0)=0$$

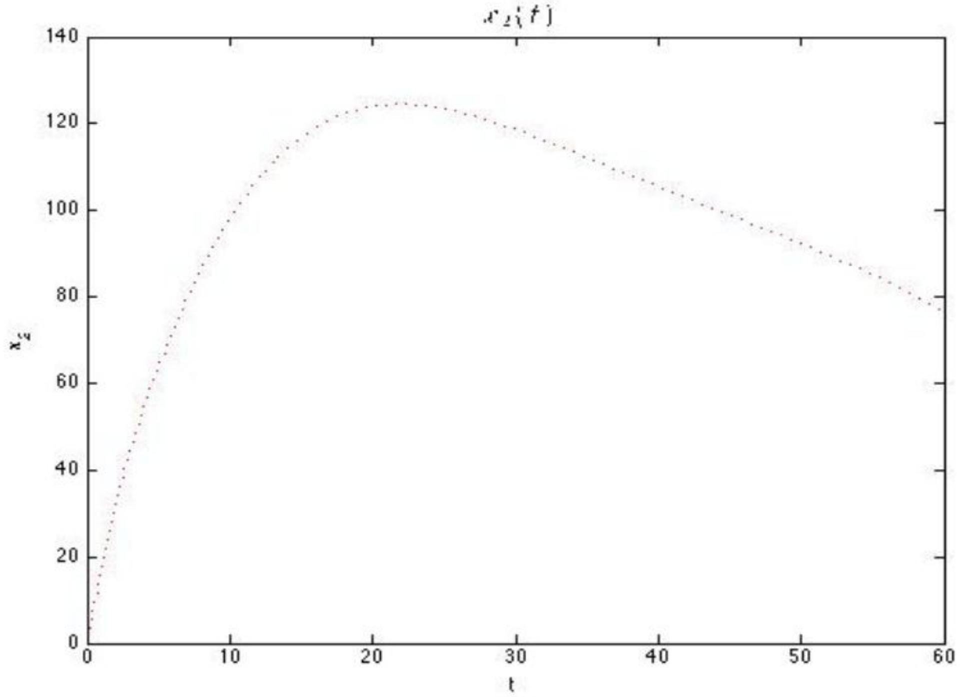
It requires the application of numerical technique of ordinary differential equation and Newton Iteration $\alpha_i=\alpha_{i-1}-\frac{\frac{\partial x_3}{\partial \alpha}}{\frac{\partial^2 x_3}{\partial \alpha^2}}$. If we want to calculate $\frac{\partial^2 x_3}{\partial \alpha^2}$, we should formally try partial derivative to obtain $\frac{\partial^2 x_i}{\partial \alpha^2}$. When it comes to higher-rise buildings, there will be more variables related to the allocation of people, where we can use Jacobian Transformation of the equation system to achieve Newton Iteration:

$$A_i=A_{i-1}-J^{-1}(E)\times E$$

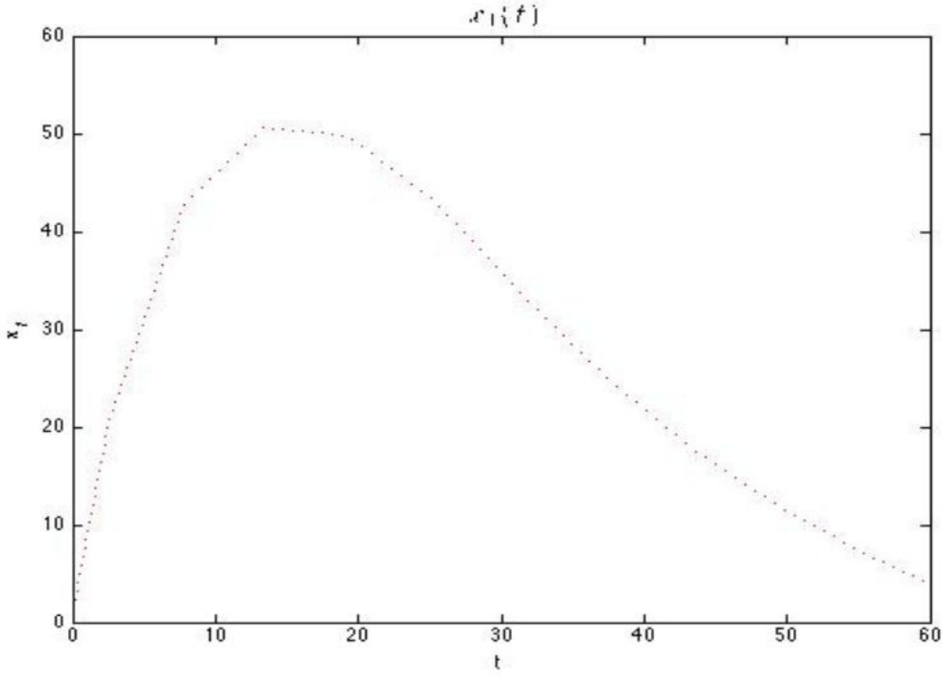
The problem is almost solved now with this tool. With the help of computer software, we can obtain the optimum $s(x)$ we want, and boundary values can be found by shooting method or cutting method.

In last autumn's mimic evacuation in our school, we regard safe time t_0 as 60 seconds. The teaching buildings in our school are three-storey buildings. Students in the first floor can quickly evacuate without going by any stairways, and the actual number of participators in the second and third floor is about 450. Therefore, we suppose $A\approx 10;\ m\approx 0.09$.

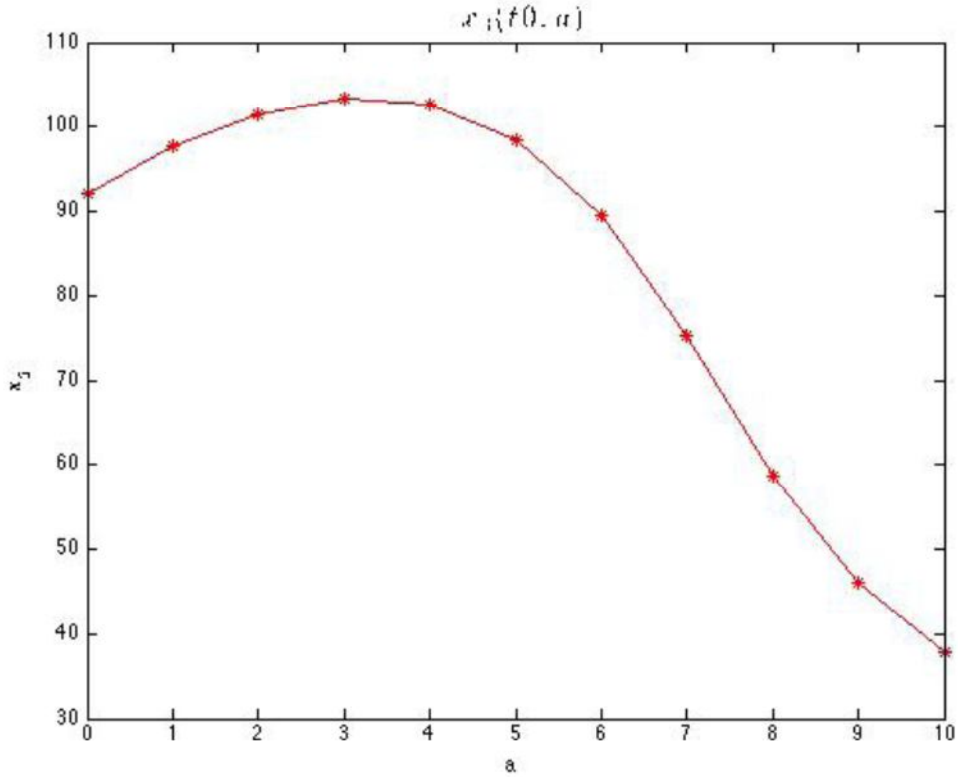
Assume that we can modify the allocation of students in the second and third floor, and we can obtain the results like that (Here we define step as 1 to do difference, and in the image, $\alpha=0$ means all the students are in the second floor, while $\alpha=10$ means all the students are in the third floor):



$x_2(t)$
x2
t



$x_1(t)$
x1
t



Through the above-mentioned method, we can find the optimum α, and corresponding $x_1(t)$, $x_2(t)$. However, we have realized that it requires a large quantity of calculation to draw an accurate phase diagram of $x_3(t_0,\alpha)$, for it means calculating every solution corresponding to a different α.(Computer code ①)

As for buildings higher than three storeys, we can use the same method to build up our model and obtain the optimum specific values of people allocation in different storeys. The only difference is the number of equations and unknown numbers.

Through this method, we have already got the conclusion, which can guide the people allocation in different storeys in high-rise buildings. In reality, we can apply this conclusion to construction devisal and space distribution.

If we want to describe the evacuation while the people distribution in the storey can be modified, the $y(t)$, which has a certain form in our model before, must be transformed into an unknown function in our new model, which only has the fixed boundary value. If we obtain the $y(t)$ corresponding to the optimum evacuation plan, we can demarcate the area in the storey corresponding to the interval $[t_1, t_2]$. It takes $t \in [t_1, t_2]$ for people in this area to reach the stairway entrances, and the aggregate number of people in this area should be arranged to approach $\int_{t_1}^{t_2} y(t)dt$.Take shopping malls, for instance, as people densities in different shops are quite different, we can encourage shopkeepers to choose shop locations and therefore make the regular people distribution approach our suggested number.

In the following model, we attempt to find the optimum evacuation plan in strong conditions, which means we can modify the people distribution in a specific storey. As the boundary conditions of calculus of variations will be involved, in the following model, we use $y(t)$ to represent the number of people who are already in the stairways, and use $y'(t)$ to represent the number of rush-into people in per unit of time:

3.3 Model in Strong Conditions

To make some further discussions, we consider about the situation that we can modify both the allocation of people in different storeys and the number of people rushing into the stairway at a certain time, which makes it a more complicated question requiring to be solved step by step. Take a common three-storey building, for example, the evacuation can be described by the following equation system:

$$\frac{dx_1}{dt} = -s(x_1) + y_1(t)$$

$$\frac{dx_2}{dt} = -s(x_2) + s(x_1) + y_2(t)$$

$$\frac{dx_3}{dt} = s(x_2)$$

This equation system can give us a direct vision that, to each defined pair of $y_1(t)$ and $y_2(t)$, there is a certain corresponding $x_3(t_0) = \int_0^{t_0} s(x_2)dt$. We can define $\frac{dx_3}{dt} = s(x_2)$, with a condition of $x_3(0) = 0$, to replace the integral form. However, both forms establish a mapping $[y_1(t), y_2(t)] \longmapsto x_3$. According to the definition of functional, this mapping is a binary functional. And what we need is the extreme value of the functional.

A simple case of this kind of problems can be expressed as:
Define $y(t) \longmapsto x(t_0)$:

$$\frac{dx}{dt} = f(x, y, y', t)$$

$$x(0) = x_0; y(0) = y_0; y(t_1) = y_1$$

Find the $y(t)$ which can make $x(t_0)$ an extreme value in its neighborhood.

To solve this question, we need to do the following steps:

3.3.1 A Single Perturbation's Influence on ODE

Firstly we discuss about the influence on the ordinary differential equation caused by a single perturbation at the initial time. Define $F(x_0, t_0)$:

$$\frac{dx}{dt} = f(x,t)$$
$$x(0) = x_0$$
$$F(x_0, t_0) = x(t_0)$$

Solve:

$$\lim_{\Delta\alpha\to 0} \frac{F(x_0 + \Delta\alpha, t_0) - F(x_0, t_0)}{\Delta\alpha}$$

To solve it, we discuss the influence on α caused by a perturbation at the same order as x:
Define $x = x' + \alpha$, the original equation can therefore be written as:

$$\frac{d(x' + \alpha)}{dt} = f(x' + \alpha, t)$$
$$x'(0) \equiv x_0$$

Same as the above-mentioned discussion, both sides of the equation can be partial derived and we can obtain:

$$\text{Left} = \frac{d}{dt}\frac{\partial(x' + \alpha)}{\partial\alpha} = \frac{d}{dt}\frac{\partial x'}{\partial\alpha}$$
$$\text{Right} = \frac{\partial f(x' + \alpha, t)}{\partial x' + \alpha} \cdot \frac{\partial(x' + \alpha)}{\partial\alpha} = \frac{\partial f(x' + \alpha, t)}{\partial(x' + \alpha)} \cdot (\frac{\partial x'}{\partial\alpha} + 1)$$

Thus,

$$\frac{d}{dt}\frac{\partial x'}{\partial\alpha} = \frac{\partial f(x' + \alpha, t)}{\partial(x' + \alpha)} \cdot (\frac{\partial x'}{\partial\alpha} + 1)$$

Take the definition of $F(x_0, t_0)$ into consideration, we have:

$$\lim_{\Delta\alpha\to 0} \frac{F(x_0 + \Delta\alpha, t_0) - F(x_0, t_0)}{\Delta\alpha} = \frac{\partial x(t_0)}{\partial\alpha}$$

Therefore, the influence on x' and x caused by a first order perturbation $\Delta\alpha$ at arbitrary moment can be expressed as:

$$\Delta x'(t_0) = \frac{\partial x}{\partial\alpha}(t_0)\times\Delta\alpha$$
$$\Delta x(t_0) = \Delta(x'(t_0) + \alpha) = (\frac{\partial x}{\partial\alpha}(t_0) + 1)\Delta\alpha$$

Define $\beta = \frac{\partial x\prime}{\partial\alpha} + 1$, we have:

$$\frac{d\beta}{dt} = \frac{\partial f(x' + \Delta\alpha, t)}{\partial(x' + \Delta\alpha)}\beta = \frac{\partial f(x,t)}{\partial x}\beta$$

$$\beta(t_0) = 1$$

We name β as amplify factor, and the change of x at any moment can be expressed as:

$$\Delta x = \beta \Delta \alpha$$

Now we can extrapolate the influence caused by the perturbation at an arbitrary moment from its counterpart at the initial moment. Define $F(x_0, t_1, t_2, \alpha)$:

$$\frac{du}{dt} = f(u,t)$$

$$u(0) = x_0$$

$$x_1 = u(t_1)$$

$$u_1 = x_1 + \alpha$$

$$\frac{dv}{dt} = f(v,t)$$

$$v(t_1) = u_1$$

$$F(x_0, t_1, t_2, \alpha) = v(t_2)$$

Solve:

$$\lim_{\alpha \to 0} \frac{F(x_0, t_1, t_2, \alpha) - F(x_0, t_1, t_2, 0)}{\alpha}$$

According to the above-mentioned discussion, we define:

$$\frac{dx}{dt} = f(x,t); x(0) = x_0$$

$$\frac{d\beta}{dt} = \frac{\partial f(x,t)}{\partial x}\beta; \beta(t_0) = 1$$

When $\alpha \to 0$,

$$\frac{du'}{dt} = f(u', t)$$

$$u'(0) = x_0 + \frac{\alpha}{\beta(t_1)}$$

We have:

$$u'(t_1) = x_1 + \alpha$$

According to uniqueness theorem:

$$u' \equiv v \quad u', v \in [t_1, +\infty)$$

Thus,

$$v(t_2) - u(t_2) = u'(t_2) - u(t_2) = \frac{\beta(t_2)}{\beta(t_1)}\alpha$$

Speaking of $\frac{dx}{dt} = f(x,t)$, when $x(t_0) = x_0 + \Delta\alpha$, where $\Delta\alpha$ is a first order perturbation, the influence on the equation is linear. Thus, a perturbation of $\Delta\alpha$ at time t_1 will be amplified to $\frac{\beta(t_2)}{\beta(t_1)}\Delta\alpha$ at moment t_2, by which we are able to calculate the influence at arbitrary moment caused by a perturbation at another

arbitrary moment, with a single $\beta(t)$.
Extrapolate it to equation system:

$$\frac{dx}{dt} = f(x, y, t)$$
$$\frac{dy}{dt} = g(x, y, t)$$

Same as the above-mentioned discussion, we define the perturbation on two directions are $\Delta\alpha_x, \Delta\alpha_y$:

$$\frac{d\beta_x}{dt} = \frac{\partial f}{\partial x}\beta_x + \frac{\partial f}{\partial y}\gamma_{xy}$$
$$\frac{d\gamma_{xy}}{dt} = \frac{\partial g}{\partial x}\beta_x + \frac{\partial g}{\partial y}\gamma_{xy}$$
$$\frac{d\gamma_{yx}}{dt} = \frac{\partial f}{\partial x}\gamma_{yx} + \frac{\partial f}{\partial y}\beta_y$$
$$\frac{d\beta_y}{dt} = \frac{\partial f}{\partial x}\gamma_{yx} + \frac{\partial f}{\partial y}\beta_y$$
$$\beta(0) = 1, \gamma(0) = 0$$

The influences on two directions at an arbitrary moment are:

$$\Delta x = \beta_x \Delta\alpha_x + \gamma_{yx}\Delta\alpha_y$$
$$\Delta y = \gamma_{xy}\Delta\alpha_x + \beta_y \Delta\alpha_y$$

Similarly, we can also calculate $\Delta\alpha_x$, $\Delta\alpha_y$ from certain Δx, Δy. The discussion above has proved that a perturbation at an arbitrary moment can be replaced by a perturbation at the initial time. Define $\gamma_{ii} = \beta_i$, and use the language of linear algebra:

$$\Gamma = \begin{pmatrix} \gamma_{11} & \cdots & \gamma_{n1} \\ \vdots & \ddots & \vdots \\ \gamma_{1n} & \cdots & \gamma_{nn} \end{pmatrix}$$

γ_{ij} means the perturbation on the direction of i's influence on the direction of j. Thus, the perturbation vector at the initial time will be amplified to:

$$\begin{pmatrix} \gamma_{11} & \cdots & \gamma_{n1} \\ \vdots & \ddots & \vdots \\ \gamma_{1n} & \cdots & \gamma_{nn} \end{pmatrix} \times \begin{pmatrix} \Delta\alpha_1 \\ \vdots \\ \Delta\alpha_n \end{pmatrix}$$

The perturbation at the moment t_1 will be amplified to what we show below at the moment t_2:

$$\Gamma(t_2) \times \Gamma^{-1}(t_1) \times \begin{pmatrix} \Delta\alpha_1 \\ \vdots \\ \Delta\alpha_n \end{pmatrix}$$

Assume that there are two perturbations $\Delta\alpha_1, \Delta\alpha_2$, which occur at different time. Keeping the generality, we make $\Delta\alpha_1$ occur at the initial time, and the equation influenced by $\Delta\alpha_2$ should be:

$$\frac{d}{dt}(x - \Delta\alpha_1) = f(x - \Delta\alpha_1, t)$$

Treat it like the equations before. When $\Delta\alpha_1 \to 0$, the equation of amplify factor will degenerate to:

$$\frac{d\beta}{dt} = \frac{\partial f(x,t)}{\partial x}\beta$$

In another word, the two perturbations' influences on each other are second order perturbations, or you can say that the two perturbations can be superimposed linearly. Also, we can find that the amplifying parts of the perturbations only depend on the constant parts, and are independent of the perturbations. Since all perturbations' contribution is still a first order perturbation, they have no influence on the next perturbation. Thus, as for the continuous second order perturbations, we can integrate them.

3.3.2 Continuous Second Order Perturbations' Influence on ODE

The second order perturbation mentioned here is produced by the variation of y, whose integration is a first order perturbation independent of the coming perturbations. If the second order perturbation is a explicit expression, we have:

$$\Delta x = \int_{t_1}^{t_2} \frac{\beta(t_2)}{\beta(t)} \Delta^2\, \alpha(t) dt$$

As for $\frac{dx}{dt} = f(x, y, y', t)$, the arbitrary variation of $y(t)$ at an arbitrary moment will cause a perturbation in dt, which is:

$$\left(\frac{\partial f}{\partial y}\delta y + \frac{\partial f}{\partial y'}\delta y'\right)dt$$

Since the equation has explicit x, such a perturbation will be amplified to:

$$\frac{\beta(t_2)}{\beta(t)}\left(\frac{\partial f}{\partial y}\delta y + \frac{\partial f}{\partial y'}\delta y'\right)dt$$

whose integral is：[4]

$$\int_{t_1}^{t_2} \frac{\beta(t_2)}{\beta(t)}\left(\frac{\partial f}{\partial y}\delta y + \frac{\partial f}{\partial y'}\delta y'\right)dt$$

$$\int_{t_1}^{t_2} \frac{\beta(t_2)}{\beta(t)}\left(\frac{\partial f}{\partial y}\delta y + \frac{\partial f}{\partial y'}\delta y'\right)dt$$

$$= \int_{t_1}^{t_2} \frac{\beta(t_2)}{\beta(t)}\frac{\partial f}{\partial y}\delta y\, dt + \int_{t_1}^{t_2} \frac{\beta(t_2)}{\beta(t)}\frac{\partial f}{\partial y'}\delta y'\, dt$$

$$= \int_{t_1}^{t_2} \frac{\beta(t_2)}{\beta(t)} \frac{\partial f}{\partial y} \delta y \, dt + \frac{\beta(t_2)}{\beta(t)} \frac{\partial f}{\partial y'} \delta y|_{t_0}^{t_1} + \int_{t_1}^{t_2} \delta y \frac{d}{dt}\left(\frac{\beta(t_2)}{\beta(t)} \frac{\partial f}{\partial y'}\right) dt$$

$$\because \delta y(t_2) = \delta y(t_1) = 0, \because \frac{\beta(t_2)}{\beta(t)} \frac{\partial f}{\partial y'} \delta y|_{t_1}^{t_2} = 0$$

$$\therefore \int_{t_1}^{t_2} \frac{\beta(t_2)}{\beta(t)} (\frac{\partial f}{\partial y} \delta y + \frac{\partial f}{\partial y'} \delta y')dt = \int_{t_1}^{t_2} \left(\frac{\beta(t_2)}{\beta(t)} \frac{\partial f}{\partial y} - \frac{d}{dt} \frac{\beta(t_2)}{\beta(t)} \frac{\partial f}{\partial y'}\right) \delta y dt$$

$\because$ the arbitrariness of δy,

$\therefore \frac{\beta(t_0)}{\beta(t)} \frac{\partial f}{\partial y} - \frac{d}{dt} \frac{\beta(t_0)}{\beta(t)} \frac{\partial f}{\partial y'} = 0$ is the requirement of the extreme value of $x(t_0)$.

We can reduce $\beta(t_0)$ here. However, we'd better keep it for formality since the transposed matrix Γ can't be reduced.

In conclusion, when $y(\mathrm{t})$ is definite, we have:

$$\frac{dx}{dt} = f(x, y, y', t)$$

$$\frac{d\beta}{dt} = \frac{\partial f(x, y, y't)}{\partial x} \beta$$

$$\frac{dy}{dt} = y'$$

whose variation is:

$$\int_{t_1}^{t_2} (\frac{\beta(t_2)}{\beta(t)} \frac{\partial f}{\partial y} - \frac{d}{dt} \frac{\beta(t_2)}{\beta(t)} \frac{\partial f}{\partial y'}) \delta y dt$$

When $x(t_0)$ comes to extreme, we have:

$$\frac{\beta_0}{\beta(t)} \frac{\partial f}{\partial y} - \frac{d}{dt} \frac{\beta_0}{\beta(t)} \frac{\partial f}{\partial y'} = 0$$

From this, we can obtain the explicit expression of $\frac{dy\prime}{dt}$, whose restrictive conditions are:

$$x(t_x) = x_0, y(t_x) = y_0, y(t_0) = y_1, \beta(t_x) = 1, \beta_0 = \beta(t_0)$$

All these make up another two-point boundary value problem.

3.3.3 Process of Model Building

According to the discussion in 3.1, we can obtain the following function:

$$s(x) = v_0 + v_1 e^{-k_1\left(\frac{x_1}{S} - \rho_1\right)^2} - \left(v_0 + v_1 e^{-k_1(\rho_1)^2}\right) e^{-\frac{k_2 x_1}{S}}$$

As our model can control the change of the quantity of rush-into people in per unit of time with the passage of time after emergencies, we should consider not only the evacuation efficiency, but also the security of our evacuation plan. Otherwise we can make all the people gather around the stairway entrance, to ensure they can rush into the stairway just after emergencies happen, ignoring the risk this action may bring about. That's why the model in 3.1, which satisfies mathematical requirement, is not actually reasonable.

In most public places, stairways have obstacles such as corners and fire prevention doors. Even for stairways directly connected to passageways, the people flow width will be cut down by a half while rushing into stairways, which may also lead to high possibility of the risk.

Consulting a treatise called ***Microscopic Modeling and Simulation Analysis of Crowd Stampede Accident Consequences***, written by Qingsong Zhang and Jinlan Liu, we adopted this definition formula from this treatise to describe the risk R_{CT}:

$$R_{CT} = \lambda_0 \left(\frac{N_s}{N_t}\right) F_N$$

In this formula, λ_0 is a parameter. N_s is the detained people's quantity at moment t. N_t is the aggregate number of people. F_N is the number of people who experience the risk (including pushing, slip, tumble etc.). [5]

Limited by our ability of calculating and investigating, we roughly consider F_N, N_s both in direct ratio to y' (the number of rush-into people in per unit of time):

$$y' \propto N_s$$
$$F_N \propto N_s 。$$

Therefore, R_{CT} is in direct ratio to y'^2. Its integral to time is the expected risk during the whole evacuation.

To better evaluate an evacuation plan, in the angle of both efficiency and security, we set up an evaluation function x_2 (x_1 is used to describe people's quantity in the stairway):

$$x_2 = \int_0^{t_0} \left(v_0 + v_1 e^{-k_1\left(\frac{x_1}{S}-\rho_1\right)^2} - \left(v_0 + v_1 e^{-k_1(\rho_1)^2}\right) e^{-\frac{k_2 x_1}{S}} - \lambda(y')^2 \right) dt$$

Change it into the form of differential:

$$\frac{dx_1}{dt} = -v_0 - v_1 e^{-k_1\left(\frac{x_1}{S}-\rho_1\right)^2} + \left(v_0 + v_1 e^{-k_1(\rho_1)^2}\right) e^{-\frac{k_2 x_1}{S}} + y'$$

$$\frac{dx_2}{dt} = v_0 + v_1 e^{-k_1\left(\frac{x_1}{S}-\rho_1\right)^2} - \left(v_0 + v_1 e^{-k_1(\rho_1)^2}\right) e^{-\frac{k_2 x_1}{S}} - \lambda(y')^2$$

$$x_1(0) = x_2(0) = 0$$
$$y(0) = 0, y(t_0) = y_0$$

Thus, what we need to do is to find the $y(t)$ which enables $x_2(t_0)$ to get the extreme value. This $y(t)$ is the optimum evacuation plan we need and can be used to guide the people distribution in a specific storey.

3.3.4 Mathematical Processing and Realization at Matlab

According to the discussion in 3.3.1, each perturbation at the initial time

$$\begin{pmatrix}\Delta\alpha_1\\ \Delta\alpha_2\end{pmatrix}$$

will be amplified to

$$\begin{pmatrix}\Delta x_1\\ \Delta x_2\end{pmatrix}=\begin{pmatrix}\gamma_{11} & \gamma_{21}\\ \gamma_{12} & \gamma_{22}\end{pmatrix}$$

Thus, the perturbation at the moment t

$$\begin{pmatrix}\Delta\alpha_1(t)\\ \Delta\alpha_2(t)\end{pmatrix}$$

is equivalent to a perturbation at $t=0$

$$\begin{pmatrix}\Delta\alpha_1(0)\\ \Delta\alpha_2(0)\end{pmatrix}=\begin{pmatrix}\gamma_{11} & \gamma_{21}\\ \gamma_{12} & \gamma_{22}\end{pmatrix}^{-1}\times\begin{pmatrix}\Delta\alpha_1(t)\\ \Delta\alpha_2(t)\end{pmatrix}$$

whose variation is

$$\int_0^{t_0}\begin{pmatrix}\gamma_{11}(t_0) & \gamma_{21}(t_0)\\ \gamma_{12}(t_0) & \gamma_{22}(t_0)\end{pmatrix}\times\begin{pmatrix}\gamma_{11} & \gamma_{21}\\ \gamma_{12} & \gamma_{22}\end{pmatrix}^{-1}\times\begin{pmatrix}\dfrac{\partial f_1}{\partial y}\delta y+\dfrac{\partial f_1}{\partial y'}\delta y'\\ \dfrac{\partial f_2}{\partial y}\delta y+\dfrac{\partial f_2}{\partial y'}\delta y'\end{pmatrix}dt$$

$$\int_0^{t_0}\sum_{i=1}^{2}\left(\begin{pmatrix}\gamma_{11}(t_0) & \gamma_{21}(t_0)\\ \gamma_{12}(t_0) & \gamma_{22}(t_0)\end{pmatrix}\times\begin{pmatrix}\gamma_{11} & \gamma_{21}\\ \gamma_{12} & \gamma_{22}\end{pmatrix}^{-1}\times\begin{pmatrix}\dfrac{\partial f_1}{\partial y}\\ \dfrac{\partial f_2}{\partial y}\end{pmatrix}\right.$$

$$\left.-\frac{d}{dt}\left(\begin{pmatrix}\gamma_{11}(t_0) & \gamma_{21}(t_0)\\ \gamma_{12}(t_0) & \gamma_{22}(t_0)\end{pmatrix}\times\begin{pmatrix}\gamma_{11} & \gamma_{21}\\ \gamma_{12} & \gamma_{22}\end{pmatrix}^{-1}\times\begin{pmatrix}\dfrac{\partial f_1}{\partial y'}\\ \dfrac{\partial f_2}{\partial y'}\end{pmatrix}\right)\right)\delta u_i\,dt$$

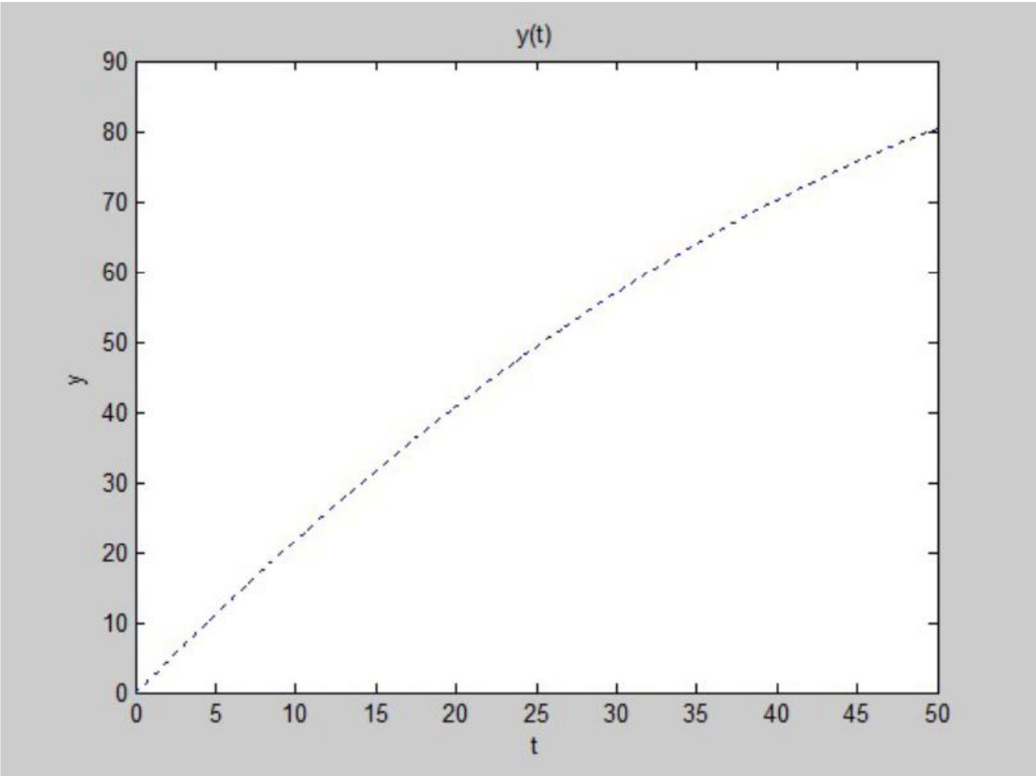
Because the arbitrariness of δu_i，to make $x_2(t_0)$ come to extreme，we have：

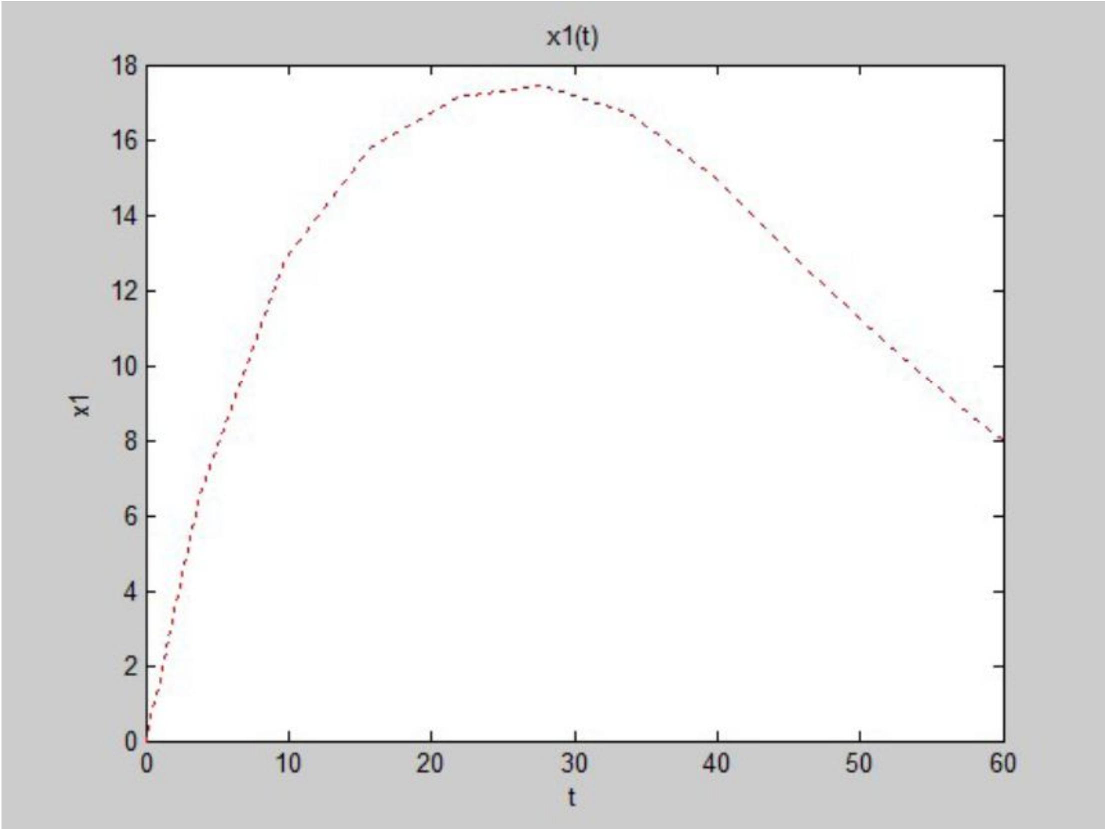
$$\begin{pmatrix}\gamma_{11}(t_0) & \gamma_{21}(t_0)\\ \gamma_{12}(t_0) & \gamma_{22}(t_0)\end{pmatrix}\times\begin{pmatrix}\gamma_{11} & \gamma_{21}\\ \gamma_{12} & \gamma_{22}\end{pmatrix}^{-1}\times\begin{pmatrix}\dfrac{\partial f_1}{\partial y}\\ \dfrac{\partial f_2}{\partial y}\end{pmatrix}$$

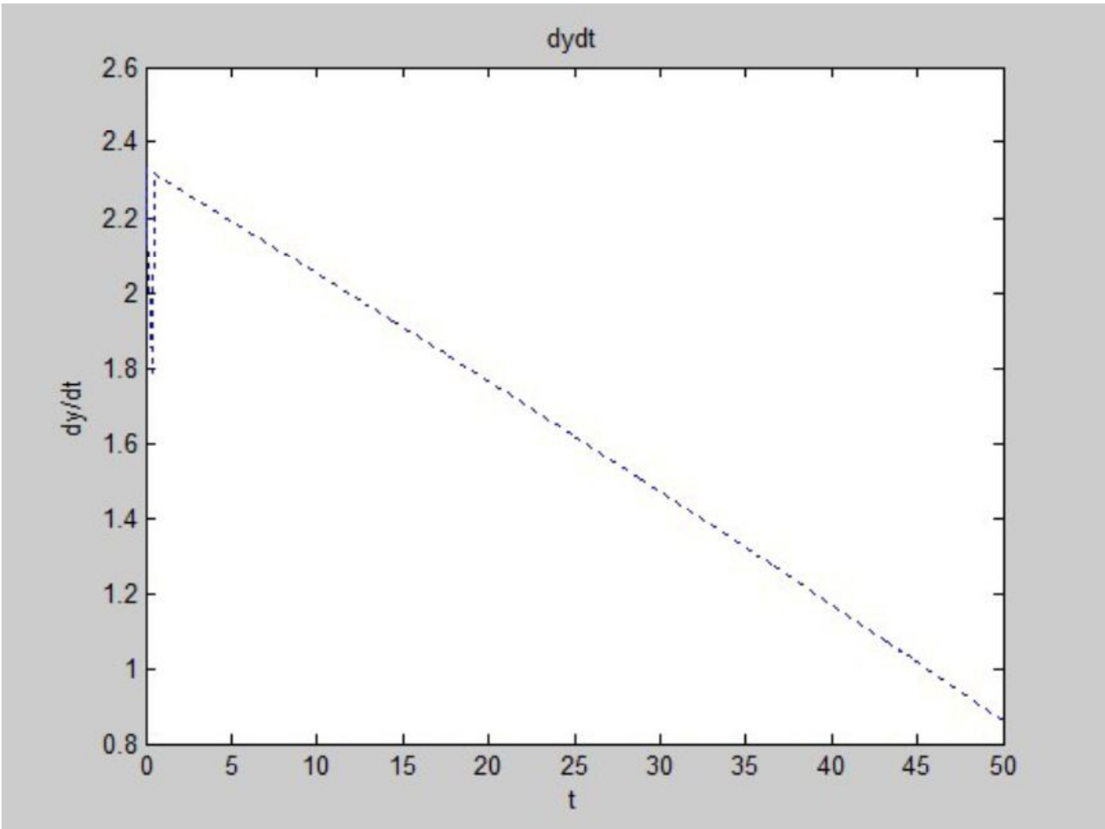
$$-\frac{d}{dt}\left(\begin{pmatrix}\gamma_{11}(t_0) & \gamma_{21}(t_0)\\ \gamma_{12}(t_0) & \gamma_{22}(t_0)\end{pmatrix}\times\begin{pmatrix}\gamma_{11} & \gamma_{21}\\ \gamma_{12} & \gamma_{22}\end{pmatrix}^{-1}\times\begin{pmatrix}\dfrac{\partial f_1}{\partial y'}\\ \dfrac{\partial f_2}{\partial y'}\end{pmatrix}\right)$$

whose second line equals 0.
Here we take a short passageway for instance, with $s = 10; v_0 = 0.182; v_1 = 1.46$. Assume it can control people distribution in the radius of 80 and the target time is 50. The aggregate number of people is 80.

The solution is:



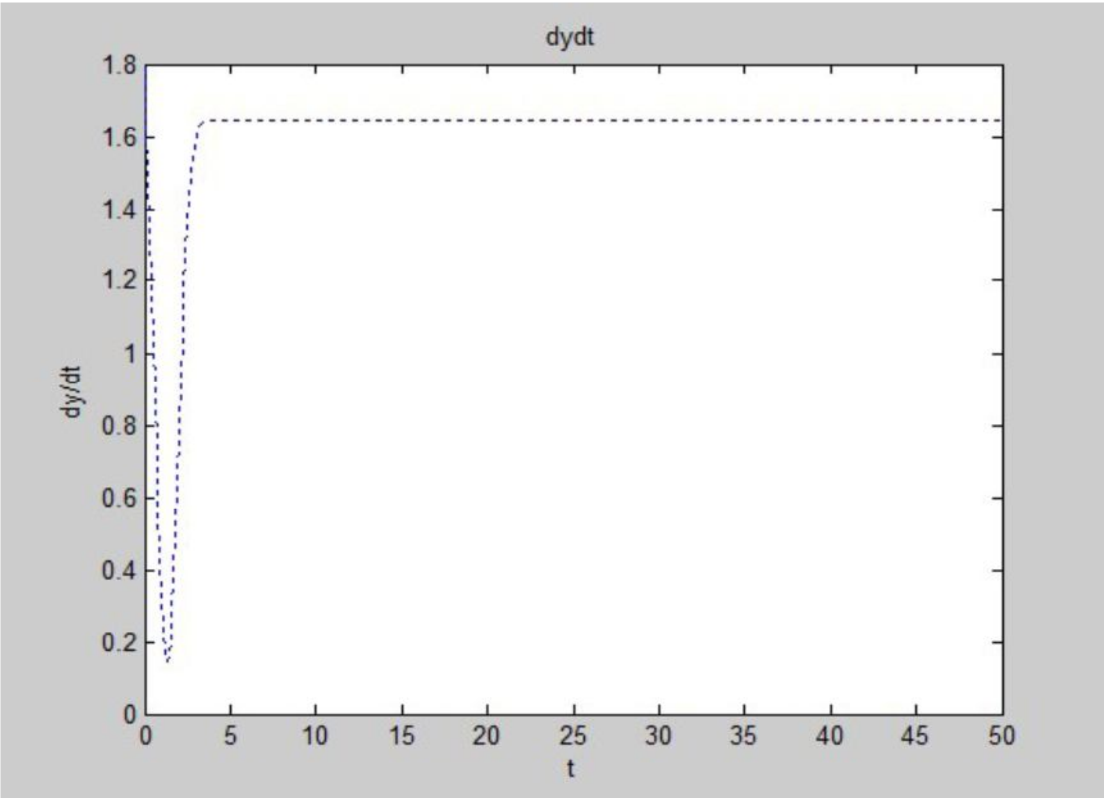

x1(t)
x1
t
0
2
4
6
8
10
12
14
16
18
0
10
20
30
40
50
60


dydt
dy/dt
t
0.8
1
1.2
1.4
1.6
1.8
2
2.2
2.4
2.6
0
5
10
15
20
25
30
35
40
45
50

The initial part of the solution oscillates fiercely, which is mainly caused by the error of programming. During modifying the initial value of iteration, the amplitude and direction of oscillation changes while the main part of the solution keeps steady. The third graph can partly instruct the allocation of people in the compass of competence in a certain exit passageway.

Like other methods based on iteration, whether it can be successfully achieved largely depends on whether the initial-value can make the iteration converge in the right direction. In addition, the parameter we need to guess includes the elements of the transposed matrix, which are difficult to estimate, so it makes a higher requirement to guessing than average iteration problems.

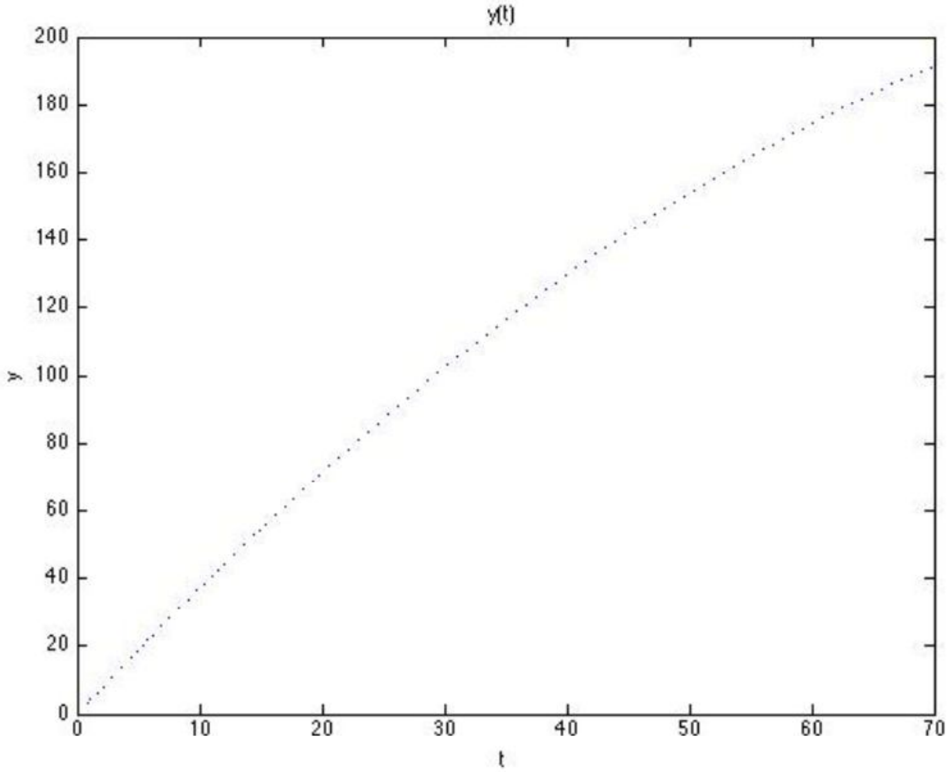
If an improper initial value is fixed, the solution may be like this:

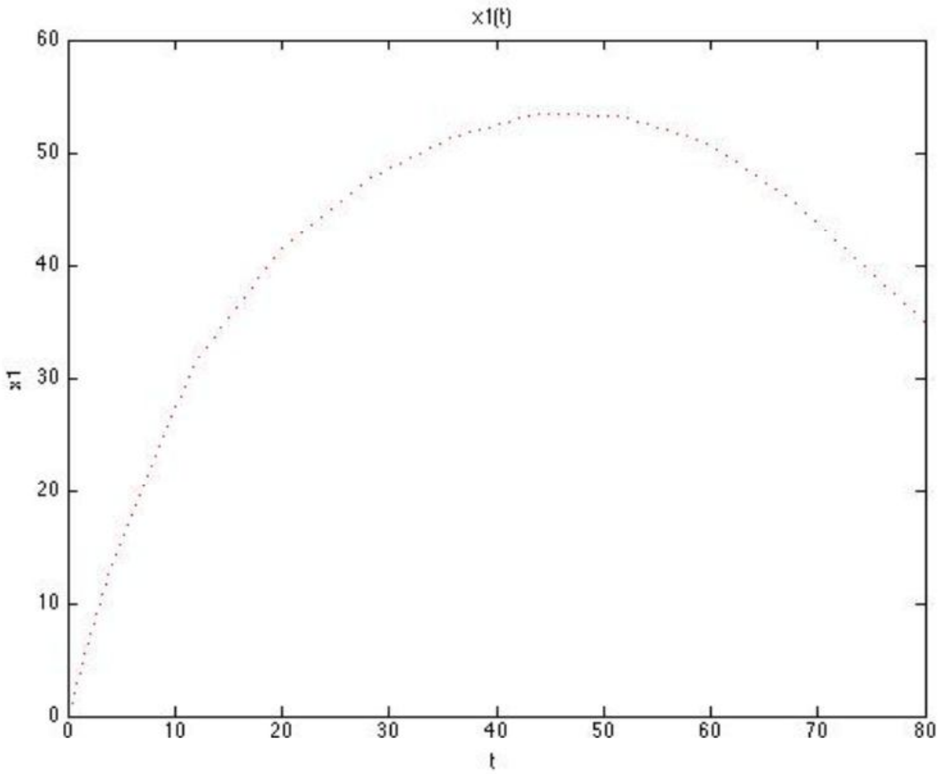


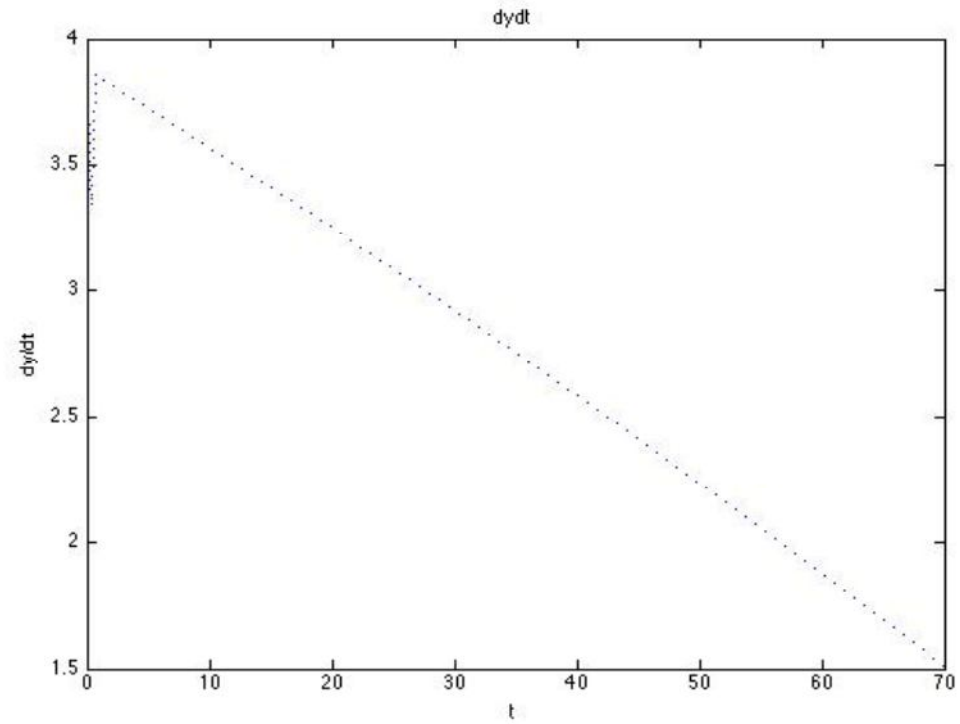
(computer code②)

The circumstance of our school is that there are four sets of stairways standing symmetrically on the four corners of the rectangle-shape building. It is obvious that the optimum solution must satisfy the symmetry of stairways. Assume that the stairways between the second floor and the third floor have much less pressure, thus we mainly considerate the pressure of the stairways between the first floor and the second floor. Therefore, the quantity of people rushing into the stairways between the first and second floor in a certain period of time can also be solved by variation problem defined by the above-mentioned equation system.

Substitute the parameter values we have obtained in 3.1:
k_1=0.6519 k_2=5.1020 ρ_1=1.760 s=26.77（obtained by measure）
v_0=0.2752 v_1=2.2079
The solution is：







(computer code③)

If we take the pressure of stairways between the second and third floor into consideration, we should add equations. Actually, as for more-than-three-store buildings, we can also use the variation theorem to obtain the solution, like the model in 3.2. The only difference is that we should add equations and unknown numbers and lead the functional modifying boundary method into our model. As for the three-storey building problem, we can list all equations and restrictive conditions:

$$\frac{dx_1}{dt} = -v_0 - v_1 e^{-k_1\left(\frac{x_1}{S}-\rho_1\right)^2} + \left(v_0 + v_1 e^{-k_1(\rho_1)^2}\right)e^{-\frac{k_2 x_1}{S}} + y_1{}'$$

$$\frac{dx_2}{dt} = -v_0 - v_1 e^{-k_1\left(\frac{x_2}{S}-\rho_1\right)^2} + \left(v_0 + v_1 e^{-k_1(\rho_1)^2}\right)e^{-\frac{k_2 x_2}{S}} + v_0 + v_1 e^{-k_1\left(\frac{x_1}{S}-\rho_1\right)^2} - \left(v_0 + v_1 e^{-k_1(\rho_1)^2}\right)e^{-\frac{k_2 x_1}{S}} + y_2{}'$$

$$\frac{dx_3}{dt} = v_0 + v_1 e^{-k_1\left(\frac{x_2}{S}-\rho_1\right)^2} - \left(v_0 + v_1 e^{-k_1(\rho_1)^2}\right)e^{-\frac{k_2 x_2}{S}} - \lambda((y_1{}')^2 + (y_2{}')^2)$$

$$x_1(0) = x_2(0) = x_3(0) = 0$$

$$y_1(0) = y_2(0) = 0; y_1(t_0) + y_2(t_0) = y_0$$

Solve y_1, y_2 when $x_3(t_0)$ comes to extreme.(To write down 3×3 transposed matrix Γ, we can use the same method as 3.3.1. We don't repeat it here.)

The corresponding Euler Equation is:

$$\Gamma(t_0)\times\Gamma^{-1}(t)\times\begin{pmatrix}\frac{\partial f_1}{\partial y_i}\\ \frac{\partial f_2}{\partial y_i}\\ \frac{\partial f_3}{\partial y_i}\end{pmatrix}-\frac{d}{dt}\left(\Gamma(t_0)\times\Gamma^{-1}(t)\times\begin{pmatrix}\frac{\partial f_1}{\partial y_i'}\\ \frac{\partial f_2}{\partial y_i'}\\ \frac{\partial f_3}{\partial y_i'}\end{pmatrix}\right), i=1,2$$

The third line of the expression is 0, which means two more equations. With $\frac{dy_i}{dt}=y_i', i=1,2$ added it becomes a dynamic system. We have 16 equations and 16 unknown functions. 9 elements, which are final values exactly, need to be assumed. The original equation system provides 6 boundary values. The transposed matrix ' initial value equaling to the unit matrix and the matrix's final value equaling to its assumed value provide 9 boundary values each. Now there are 24 boundary conditions in all. And the last one is determined by the functional modifying boundary value. Since there is a restrictive condition that $y_1(t_0)+y_2(t_0)=y_0$, the two variation at the final value are dependent by each other. We have:

$$\delta\begin{pmatrix}x_1\\ x_2\\ x_3\end{pmatrix}=\Gamma(t_0)\times\Gamma^{-1}(t)\begin{pmatrix}\frac{\partial f_1}{\partial y_1'}\\ \frac{\partial f_2}{\partial y_1'}\\ \frac{\partial f_3}{\partial y_1'}\end{pmatrix}\delta y_1(t_0)+\Gamma(t_0)\times\Gamma^{-1}(t)\begin{pmatrix}\frac{\partial f_1}{\partial y_2'}\\ \frac{\partial f_2}{\partial y_2'}\\ \frac{\partial f_3}{\partial y_2'}\end{pmatrix}\delta y_2(t_0)$$

The third line equals 0.
Considering $y_1(t_0)+y_2(t_0)=y_0$, or you can say $\delta y_1(t_0)+\delta y_2(t_0)=0$, we can say that the last boundary condition is:

$$\Gamma(t_0)\times\Gamma^{-1}(t)\begin{pmatrix}\frac{\partial f_1}{\partial y_1'}\\ \frac{\partial f_2}{\partial y_1'}\\ \frac{\partial f_3}{\partial y_1'}\end{pmatrix}-\Gamma(t_0)\times\Gamma^{-1}(t)\begin{pmatrix}\frac{\partial f_1}{\partial y_2'}\\ \frac{\partial f_2}{\partial y_2'}\\ \frac{\partial f_3}{\partial y_2'}\end{pmatrix}$$

The third line equals 0.(The two matrixes here are reciprocal. We keep them for formality in case we meet modifying initial value problems next time.)

In a word, to obtain the optimum plan by variation theorem is theoretically practicable. Like the model in 3.2, when more variable quantities and functions are added, variable difference method needs to difference at each dimension and the error will grow geometrically. Our method can make up the problem. Though it still involves much compute, it will be much better than variable-difference method or trying functions without destinations.

Chapter Ⅳ Project Summary

We expect to seek a universal solution to this kind of problems, and attempt to explore new ways of quantitative research on evacuation by building this model. In terms of that, we have achieved our goal.

This model can guide the action to arrange a reasonable quantity of people in a certain area, from which it will take people a certain period of time to reach stairway entrances. Therefore, it can be used to direct the distribution of functional areas in high-rise buildings, which can, to a great extent, relieve the evacuation command pressure and lessen the danger caused by potential command mistakes.

However, we must admit that this model has some insufficiencies. For instance, limited by ability of data acquisition, parameters' values are quite rough, especially in terms of risk evaluation. If this model can be combined with a more precise risk estimation model, the result can be much more accurate.

As for our prospect, the safe time t_0 can be described more precisely. As the safe time of a building cannot be estimated accurately, we once think of using probability function to describe its possible value. However, because of the lack of time, we don't achieve it, so we expect optimization in this term. Moreover, combination of our model and functional modifying boundary method can determine the people allocation in different storeys and the people distribution is a certain storey at the same time. We have made some attempt, which is mentioned at the end of our article.

References

[1] Sanbing Li, Feng Chen, Chenglei Li. Study on The Passenger Flow Density and Speed in The Subway Platform Distributed Area. Study of City Track Traffic, China, No.12 in 2009
[1.1]Paul, J.L. Effective-Width model for crowd evacuation flow on stairs[A]. Proceedings of 6th International Fire Protection Engineering Seminar[C]. Karlsruhe, Germany, 1982. 295-306.
[1.2]Pauls, J.L. Building Evacuation: Research Findings and Recommendations. Fires and Human Behaviour[M]. New York: John Wiley and Sons, 1980. 251-275.
[1.3]Smith, R.A.Density, velocity and flow relationships for closely packed crowds[J]. Safety Science,1995, 18(4),321-327.
[2]Fruin, JJ. Pedestrian planning and design[M]. New York: Metropolitan Association of Urban Designers and Environmental Planners, 1971.
[3]Ping Rao, lizhong Yang, Kongjin Zhu, Taolin Zhang. Study on The Characteristics of The Typical Student Crowd Evacuation in College under The State of Emergency. National Key Fire Science Laboratory in University of Science and Technology of China, Hefei, 230026
[4]Chongshi Wu. Methods of Mathematical Physics, Peking University Publishing House
[5]Qingsong Zhang, Jinlan Liu, Guomin Zhao. Microscopic Modeling and Simulation Analysis of Crowd Stampede Accident Consequences. Journal of Safety and Environment, August, 2008

Appendix

Computer Code:

①Model of People Allocation of The Second and Third Floor in Three-storey Buildings:

```
function r = bf3

k1 = 0.6519;
rou1 = 1.7600;
s = 26.7728;
v0 = 0.2752;
v1 = 2.2079;

t0 = 60;

m = 0.09;
A = 10;
lambda = 0:.1:1;
a = A*lambda;

  function x3t0 = ff(alpha)
      %%
      sfun = @(x) v0 + v1.*exp(-k1.*(x./s - rou1).^2);

      figure(1);
      plot(0:100,sfun(0:100),'k:');
      xlabel('$\mathrm{x}$','fontsize',15, 'interpreter','latex');
      ylabel('$\mathrm{s}$','fontsize',15, 'interpreter','latex');

title('$\mathrm{s}{(x)}$','interpreter','latex','fontsize',15);

          function dx1 = Dx1DtSubFun(t, x1)
             dx1 = -sfun(x1) + alpha.*exp(-m*t);
          end

      tspan = 0:60;
      %tspan = [0 60];
      options = odeset('RelTol', 0.0001);
      sol = ode45(@Dx1DtSubFun, tspan, 0, options);

      t45 = sol.x;
      x45 = sol.y;
```

```
        figure(2);clf; h1 = axes; %hold on;
        plot(h1, t45, x45, 'r:');
        xlabel('\itt'); ylabel('\itx_1'); title('$x_1(t)$',
'interpreter', 'latex', 'fontsize', 15);

        h = .1;step
        tneed = 0:h:t0;
        [valx1,derx1] = deval(sol, tneed);

        g = derx1 + A*exp(-m*tneed);
        %R-K
        x2 = zeros(length(tneed),1);
        for i = 1:length(tneed)-1
            K1 = h*(-sfun(x2(i)) + g(i));
            K2 = h*(-sfun(x2(i)+ K1/2) + (g(i)+g(i+1))/2 );
            K3 = h*(-sfun(x2(i)+ K2/2) + (g(i)+g(i+1))/2 );
            K4 = h*(-sfun(x2(i)+ K3) + g(i+1));
            x2(i+1) = x2(i) +  1/6*(K1 + 2*K2 + 2*K3 + K4);
        end

        figure(3);
        plot(tneed, x2, 'r:');
        xlabel('\itt'); ylabel('\itx_2'); title('$x_2(t)$',
'interpreter', 'latex', 'fontsize', 15);

        %integralx3|t0
        figure(5)
        plot(tneed, sfun(x2));
        x3t0 = h*sum( sfun(x2(1:end-1))+0.5*diff(sfun(x2)) );
    end


x3 = arrayfun(@ff,a);
figure(4);
plot(a, x3,'r*-');
xlabel('\ita'); ylabel('\itx_3'); title('$x_3(t0,a)$',
'interpreter','latex', 'fontsize', 15);
end
```

②Test of Model in Strong Conditions:

```
function r = bf
lambda = 0.015;
k1 = 0.6519;
k = 5.1020;
```

```
rou1 = 1.76;
s = 26.7728;
v0 = 0.6137;
v1 = 4.9234;

%%
t0 = 70;
yt0 = 300;
step = 0.1;%integral step
Num = 3;%

%%initial value;
x10 = 0;
x20 = 0;
y0 = 0;

%%
m1 = 2*k1*v1/s;
m2 = k/s*(v0+v1*exp(-k1*rou1^2));


%Df = df1/dx1（partial derivative）
Df = @(x1)m1.*(x1./s-rou1).*exp(-k1.*(x1./s-rou1).^2) -
m2.*exp(-k./s.*x1);
figure(3);
plot(0:100,Df(0:100),'k:');xlabel('x1'); ylabel('Df');
title('df1/dx1');

%%
   function dx1 = Dx1DtSubFun(t, x1)
      dx1 =
-v0-v1*exp(-k1*(x1/s-rou1).^2)+(v0+v1*exp(-k1*rou1^2))*exp(-k*x1/s);
      dy = zeros(size(t));
      if length(t)>1
         for j = 1:length(t)
            if t(j) < 1
               dy(j) = -(beta(t0)*exp(-t(j)*Df(0.5*t(j))) + c2) /
(2*lambda);
            else
               dy(j) = -(beta(t0) /beta(t(j))+c2)/(2*lambda);
            end
         end
      else
         if t < 1
```

```
            dy = -(beta(t0)*exp(-t*Df(0.5*t)) + c2) / (2*lambda);
        else
%               if isnan(beta(t)) && ~isnan(beta(t0))
%                   disp(['t:',num2str(t)]);
%                   disp(['beta(t):',num2str(beta(t)),'
beta(t0):',num2str(beta(t0))]);
%               end
            dy = -(beta(t0) /beta(t)+c2)/(2*lambda);
        end
    end
    dx1 = dx1 + dy;
end




epsilon = 10^(-3);
beta = @(t)exp(-0.003*t);%iteration
betafirst = beta;

tspan = [0, t0+10];
x10 = 0;

for kk = 1:2
    for k = 1:Num

    tao = 0:step:t0;
    tmp = 0;
    for i = 1:length(tao)-1
        if tao(i)<1%to get rid of warning
            tmp = tmp + step*exp(-tao(i)*Df(0.5*tao(i)));
        else
            tmp = tmp + step*1/beta(tao(i));
        end
    end
    integ = tmp;%integral

    c2 = (-beta(t0) * integ - 2*lambda*yt0 )/t0;
    %c2 = c2 - 1

    dy0 = -1/(2*lambda)*(beta(t0)+c2);
    dyt0 = -(1+c2)/(2*lambda);
    dx1t0 = dyt0 - lambda*dyt0^2;
```

```
    options = odeset('RelTol', 0.001);
    sol = ode45(@Dx1DtSubFun, tspan, x10, options);
    t45 = sol.x;
    x45 = sol.y;

    figure(1);clf; h1 = axes; %hold on;
    figure(2);clf; g1 = axes; %hold on;
    plot(h1, t45, x45, 'r:'); xlabel(h1, 't'); ylabel(h1, 'x1'); title(h1,
'x1(t)');
    plot(g1, x45, Dx1DtSubFun(t45, x45), 'b-'); xlabel(g1, 'Î»ÒÆx1');
ylabel(g1, 'ËÙ¶Èdx1'); title(g1, 'x1µÄÏàÆ½Ãæ');



    [x45t0,dx45t0] = deval(sol, t0);

    %%
    dy = @(tt)-(beta(t0)/beta(tt) + c2) / (2*lambda);
    z = 0:step:t0;
    y = zeros(size(z));
    dydt = zeros(size(z));
    for i = 1:length(z)
        if z(i)<.5%to get rid ofwaring
            dydt(i) = -(beta(t0)*exp(-z(i)*Df(0.5*z(i))) + c2) /
(2*lambda);
        else
            dydt(i) = dy(z(i));
        end

        if i==1
            y(1) = 0;
            continue;
        end
        y(i) = y(i-1)+ step*0.5*(dydt(i)+dydt(i-1));%integral
    end

    %show y(t)
    figure(4);
    plot(z, y, 'b:');xlabel('t'); ylabel('y'); title('y(t)');
    %show y'(t)
    figure(5);
    plot(z, dydt, 'b:');xlabel('t'); ylabel('dy/dt'); title('dydt');
```

```
    %%
    if kk == 1
        dydtsquare = dydt.^2;
        x2t0tmp = -x45t0 + yt0 - lambda * step * sum(dydtsquare(1:end-1)
+ 0.5*diff(dydtsquare))

        if k == 1
            x2t0 = x2t0tmp;
            bestnum = k;
        end

        if x2t0 < x2t0tmp & x2t0tmp > 0
           x2t0 = x2t0tmp;
           bestnum = k;
        end
    end
    %%
    betatmp = @(t)Df(deval(sol, t));
    beta = @(tt)exp(quadl(betatmp, 0, tt));
    end

    if kk == 1
        Num = bestnum;
        disp(['bestnum:',num2str(bestnum)]);
        beta = betafirst;%
    end
end



end
```

③Model in Strong Conditions' Functional Method's Application Test in Our school

```
function r = bf2
lambda = 0.04;
k1 = 0.6519;
k = 5.1020;
rou1 = 1.76;
s = 26.7728;
v0 = 0.2752;
v1 = 2.2079;

%%guess
```

```
t0 = 70;
yt0 = 191;
step = 0.1;%integral step
Num = 3;%iteration number

%%initial value;
x10 = 0;
x20 = 0;
y0 = 0;

%%
m1 = 2*k1*v1/s;
m2 = k/s*(v0+v1*exp(-k1*rou1^2));


%Df = df1/dx1 (partial derivative)
Df = @(x1)m1.*(x1./s-rou1).*exp(-k1.*(x1./s-rou1).^2) -
m2.*exp(-k./s.*x1);
figure(3);
plot(0:100,Df(0:100),'k:');xlabel('x1'); ylabel('Df');
title('df1/dx1');

%%
    function dx1 = Dx1DtSubFun(t, x1)
        dx1 =
-v0-v1*exp(-k1*(x1/s-rou1).^2)+(v0+v1*exp(-k1*rou1^2))*exp(-k*x1/s);
        dy = zeros(size(t));
        if length(t)>1
            for j = 1:length(t)
                if t(j) < 1
                    dy(j) = -(beta(t0)*exp(-t(j)*Df(0.5*t(j))) + c2) /
(2*lambda);
                else
                    dy(j) = -(beta(t0) /beta(t(j))+c2)/(2*lambda);
                end
            end
        else
            if t < 1
                dy = -(beta(t0)*exp(-t*Df(0.5*t)) + c2) / (2*lambda);
            else
%               if isnan(beta(t)) && ~isnan(beta(t0))
%                   disp(['t:',num2str(t)]);
%                   disp(['beta(t):',num2str(beta(t)),'
beta(t0):',num2str(beta(t0))]);
```

```
%               end
              dy = -(beta(t0) /beta(t)+c2)/(2*lambda);
          end
      end
      dx1 = dx1 + dy;
  end


epsilon = 10^(-3);
beta = @(t)exp(-0.003*t);%guess initial value as -0.0001;iteration
betafirst = beta;

tspan = [0, t0+10];
x10 = 0;

for kk = 1:2
  for k = 1:Num

  tao = 0:step:t0;
  tmp = 0;
  for i = 1:length(tao)-1
      if tao(i)<1%to get rid of warning
          tmp = tmp + step*exp(-tao(i)*Df(0.5*tao(i)));
      else
          tmp = tmp + step*1/beta(tao(i));
      end
  end
  integ = tmp;%integral

  c2 = (-beta(t0) * integ - 2*lambda*yt0 )/t0;
  %c2 = c2 - 1

  dy0 = -1/(2*lambda)*(beta(t0)+c2);
  dyt0 = -(1+c2)/(2*lambda);
  dx1t0 = dyt0 - lambda*dyt0^2;


  options = odeset('RelTol', 0.001);
  sol = ode45(@Dx1DtSubFun, tspan, x10, options);
  t45 = sol.x;
  x45 = sol.y;

  figure(1);clf; h1 = axes; %hold on;
  figure(2);clf; g1 = axes; %hold on;
```

```
    plot(h1, t45, x45, 'r:'); xlabel(h1, 't'); ylabel(h1, 'x1'); title(h1, 
'x1(t)');
    plot(g1, x45, Dx1DtSubFun(t45, x45), 'b-'); xlabel(g1, 'Î»ÒÆx1'); 
ylabel(g1, 'ËÙ¶Èdx1'); title(g1, 'x1µÄÏàÆ½Ãæ');


    [x45t0,dx45t0] = deval(sol, t0);

    %%
    dy = @(tt)-(beta(t0)/beta(tt) + c2) / (2*lambda);
    z = 0:step:t0;
    y = zeros(size(z));
    dydt = zeros(size(z));
    for i = 1:length(z)
        if z(i)<.5%get rid ofwaring
            dydt(i) = -(beta(t0)*exp(-z(i)*Df(0.5*z(i))) + c2) / 
(2*lambda);
        else
            dydt(i) = dy(z(i));
        end

        if i==1
            y(1) = 0;
            continue;
        end
        y(i) = y(i-1)+ step*0.5*(dydt(i)+dydt(i-1));%积分
    end

    %showy(t)
    figure(4);
    plot(z, y, 'b:');xlabel('t'); ylabel('y'); title('y(t)');
    %showy'(t)
    figure(5);
    plot(z, dydt, 'b:');xlabel('t'); ylabel('dy/dt'); title('dydt');


    %%
    if kk == 1
        dydtsquare = dydt.^2;
        x2t0tmp = -x45t0 + yt0 - lambda * step * sum(dydtsquare(1:end-1) 
+ 0.5*diff(dydtsquare))

        if k == 1
            x2t0 = x2t0tmp;
```

```
            bestnum = k;
        end

        if x2t0 < x2t0tmp & x2t0tmp > 0
           x2t0 = x2t0tmp;
           bestnum = k;
        end
    end
    %%
    betatmp = @(t)Df(deval(sol, t));
    beta = @(tt)exp(quadl(betatmp, 0, tt));
    end

    if kk == 1
        Num = bestnum;
        disp(['bestnum:',num2str(bestnum)]);
        beta = betafirst;
    end
end

end
```