THE DESIGN OF SELF-DEVELOPMENTAL MODEL OF ATMOSPHERIC POLLUTANT DISPERSION

Radim Dvořák[1], **Jiří Krajíček**[1], **Jaroslav Zendulka**[1], **František Zbořil**[1]

[1]Brno University of Technology, Faculty of Information Technology
612 66 Brno, Božetěchova 2, Czech Republic

idvorak@fit.vutbr.cz(Radim Dvořák)

Abstract

This paper presents the design of self-developmental model of atmospheric pollutant dispersion. As many of existing physical models of atmospheric phenomena are statically defined for certain conditions, for simulation it is emergent to update these models to reflect the natural dynamic behaviour of self-adaption. In recent research we have already proposed new evolutionary method denoted as Content/Form computing. Here we apply this evolutionary method as main iterative process to develop adaptive model of atmospheric pollutant dispersion. Main inspiration of our design comes from existing equation of atmospheric pollutant dispersion, its inner parts like wind or diffusion models and from and Sheldrake's extended evolutionary theory of "*Nature as Alive*" [1].

Keywords: pollutant dispersion, advection, dispersion, atmosphere, evolution, Content/Form computing, morphic fields.

Presenting Author's Biography

Radim Dvořák is a Ph.D. student in the Department of Intelligent Systems at the Faculty of Information Technology, Brno University of Technology, Czech Republic. He received a master degree in information technology program from the Brno University of Technology on subject of deformable bodies' dynamics. He is interested in modeling, simulation, biometric systems and computer graphics. For more information see please http://www.fit.vutbr.cz/ idvorak.

1 Introduction

There exist many models for simulation of wind flow, diffusion models, dispersion models. In the past these models were especially designed for the best certain conditions. On the other hand in real nature these ”models” are not conditionally (static) defined but are evolved (developed) across time and space.

Therefore here we design and implement the evolutionary developmental model of atmospheric pollutant dispersion. Different models are commonly evaluated by sets of experiments. Main disadvantage of these experiments is small amount of of observed data. For example only 3 measurements for several kilometric space.

Assuming the classical evolutionary methods (genetic algorithms, genetic programming), where operators (crossover, mutation) are excluded from evolution itself, due to the small amount of data it is hard to predict the model behaviour in every space location. Hence we assume and apply the Content/Form computing for the model development [2], which is the new method in evolutionary systems inspired by Sheldrake’s theory of ”*The Nature as Alive*” [1]. In this approach evolutionary operators (methods) - together with data (Form) are part of the evolution process and past developmental states are stored in the evolution memory (Content).

The structure of our paper is organized as follows. In the next section the description of the general atmospheric pollution model is described and the specific models of the wind and the turbulent diffusion is outlined. In section 3 the evolution of the content/form computing is presented including its kind of adaptation to the atmospheric model. The implementation details are included in the section 4. The final conclusion and the future work is then stated in the last section 5.

2 The Equation of Atmospheric Pollutant Dispersion

The modeling of atmospheric pollution has a great importance in the case of prediction of pollutant behavior throw the atmosphere. It has gone big journey in the last century as is stated in [3]. One of the first models was so called Gaussian plume model which was first proposed by Pasquille and others in sixties of 20th century. It was firstly used for modeling pollution dispersion from a point source, however, it was soon applied to line and area sources as well. It matched with the observations and became a standard in every industrial country. The concern shifted to studying of dispersion phenomena, namely dry dispersion. It means that the pollutant behavior above the ground and a study of its causalities had been in the area of interest. Consequently, a new full form of atmospheric equation was formulated, which contains diffusivity, advection, deposition, emission and chemistry terms. Since that time no new general model was proposed and it is focused until now to solve the full atmospheric equation:

$$\begin{aligned} \frac{\partial c}{\partial t} + \nabla V c = \nabla(D \nabla c) + \\ chemistry + emissions + \\ drydeposition + wetdeposition \end{aligned} \tag{1}$$

where the c is concentration, V is wind velocity, D is a diffusion coefficient and t is time. The chemistry and emissions terms are often marked as the reaction term, e.i. how the pollutant behaves in chemical environment. The last two terms are the major sink terms in the model and besides they determine the pollutant behaviour above the surface. The coefficient we are dealing is the velocity field which is often described as a wind model.

The general equation 1 is not used in whole form. In the models the equation is split into several smaller parts and is solved separately. It is commonly used in numerical solution of the equation; the technique is known as operator splitting. In our case we use the advection (wind) and diffusion parts of the equation. Some of the used models are outlined in the next two subsections.

2.1 Examples of Wind Models

There exist several wind models that describes the wind flow in atmosphere. One of the simpler model is one presented by Lin [4]:

$$U(z) = U(z_r)\left(\frac{z}{z_r}\right)^{\alpha} - az^{\alpha}, a = \frac{U(z_r)}{z_r^{\alpha}} \tag{2}$$

where $U(z_r)$ is measured wind speed at a referenced height and a, α are constants that depend on atmospheric stability and surface roughness [4].

Next wind model we used as the input is the one presented by Wortmann [5]. The special coefficients are needed in order to calculate the wind speed at given position. These are Monin-Obukhov length L, roughness length z_0, von Karman constant k and friction velocity $u_\star$. Then, the wind velocity is given by:

$$U(z) = u_\star\left[\ln\frac{z}{z_0} - \Psi_m\frac{z}{L} + \Psi_m\frac{z_0}{L}\right], z \le z_b \tag{3}$$

$$U(z) = U(z_b), z > z_b, \tag{4}$$

where the coefficient $z_b = min(|L|, 0.1h)$, where h is a height of the unstable boundary layer.

The function Ψ_m is of the form:

$$\begin{aligned} \Psi_m = 2\ln\left(\frac{1+A}{2}\right) + \ln\left(\frac{1+A^2}{2}\right) \\ -2\tan^{-1}(A) + \frac{\pi}{2} \end{aligned} \tag{5}$$

$$A = \left[1 - \frac{16z}{L}\right]^{0.25} \tag{6}$$

The last wind model is represented by equation of the form [6]:

$$U(z) = \frac{u_\star}{k}\{\ln\frac{z}{z_0} - \left[1 - 6.9\frac{h}{L}\right] + \frac{z - z_0}{L} - \frac{6.9h}{2L}\left[\frac{z^2}{h^2} - \frac{z_0^2}{h^2}\right]\}, \qquad (7)$$

$$\text{where} \quad h/L > 0$$

$$U(z) = \frac{u_\star}{k}\{\ln\frac{z}{z_0} + \ln\left[\frac{(1+\mu_0^2)(1+\mu_0)^2}{(1+\mu_0^2)(1+\mu_0)^2}\right] + 2[\tan^{-1}\mu - \tan^{-1}\mu_0 - \frac{2L}{33h}[\mu^3 - \mu_0^3]\}, \qquad (8)$$

$$\text{where} \quad h/L < 0$$

with

$$\mu = \left(1 - 22\frac{hz}{Lh}\right)^{0.25},$$

$$\mu_0 = \left(1 - 22\frac{hz_0}{Lh}\right)^{0.25}.$$

The variables have the same meaning as in equation 4.

2.2 Examples of Diffusion Models

Similarly to wind models the different kinds of diffusion models are used to simulate the diffusion/turbulence process in the atmosphere. The first model is represented by [6]:

$$K(z) = ku_{*o}h\frac{z}{h}\left(1 - \frac{z}{h}\right)\left(1 + 9.2\frac{h}{L}\frac{z}{h}\right)^{-1} \qquad (9)$$

$$\text{where} \quad h/L > 0$$

$$K(z) = ku_{*o}h\frac{z}{h}\left(1 - \frac{z}{h}\right)\left(1 - 13\frac{h}{L}\frac{z}{h}\right)^{1/2} \qquad (10)$$

$$\text{where} \quad h/L < 0$$

The variables has the same meaning as in equation 4.

The vertical eddy diffusivity derived by [7] is another and last presented example:

$$\frac{K(z)}{w_* z_i} = 0.22\frac{z}{h}^{1/3}\left(1 - \frac{z}{h}\right)^{1/3} \left[1 - \exp\left(-\frac{4z}{h}\right) - 0.0003\exp\left(\frac{8z}{h}\right)\right] \qquad (11)$$

The variables has the same meaning as in equation 4.

3 State of the Art: the Model Design

In previous section we have schematically described the main parts of the equation for atmospheric pollutant dispersion, especially wind models. Here we describe how these physical models are involved in Content/Form computing evolutionary process to create one self-developmental compact model.

Firstly we briefly recall the Content/Form computing and then apply this system on atmospheric pollutant dispersion development by describing the Form and Content components.

3.1 The Content/Form Computing

In recent work [2] we have proposed the method of Content/Form computing. In comparison with present methods in evolutionary computing (GA, GP) [8], [9] its is similar but different computing process. This is mostly caused by considering the radical Sheldrake's extended evolutionary theory [1] instead of classical Darwinian assumption. As new type of evolutionary computing, it can be applied on various AI problems (e.g. finding the optimal solution) or as the developmental system on various models (model adaptation) – this case.

Content/Form Computing operates with abstract terms of the Content and the Form. The Content encapsulates the Form which is being evolved according to Content Memory and Goals. Meanwhile the Content consists of the memory of past states, goals and fitness function (evaluation), the Form includes input/output data as population and set of methods as operators. Since methods itself are also instances of the Form – are also evolved, beside the methods for the data development we also operate with methods for methods development.

In following subsections we consequently describe each components of the Content and Form in the context of application on equation of atmospheric pollutant dispersion development.

3.2 FORM: Methods and Data

First instances of the Form are methods, which can be further categorised to methods for methods development and methods for data development, see Figure 1. In our case set of methods for data development includes:

- The "*growth method*" – advection diffusion equation (Form1.1 in Figure 1).
- Methods for data-in development (Form1.3).

Further the set of methods for methods (Form1.2 in Figure 1) development contains:

- Method for model's operators evolution (for smooth changes in certain variant of wind or diffusion model).
- Method for switching among variants of models (for rough changes in wind or diffusion models).

The data are represented by data-in (Form0.0 in Figure 1) which stands for experiment setting, conditions and other constants for atmospheric pollutant dispersion equation and by data-out (Form0.1) which stands for the expected solution – concentration.

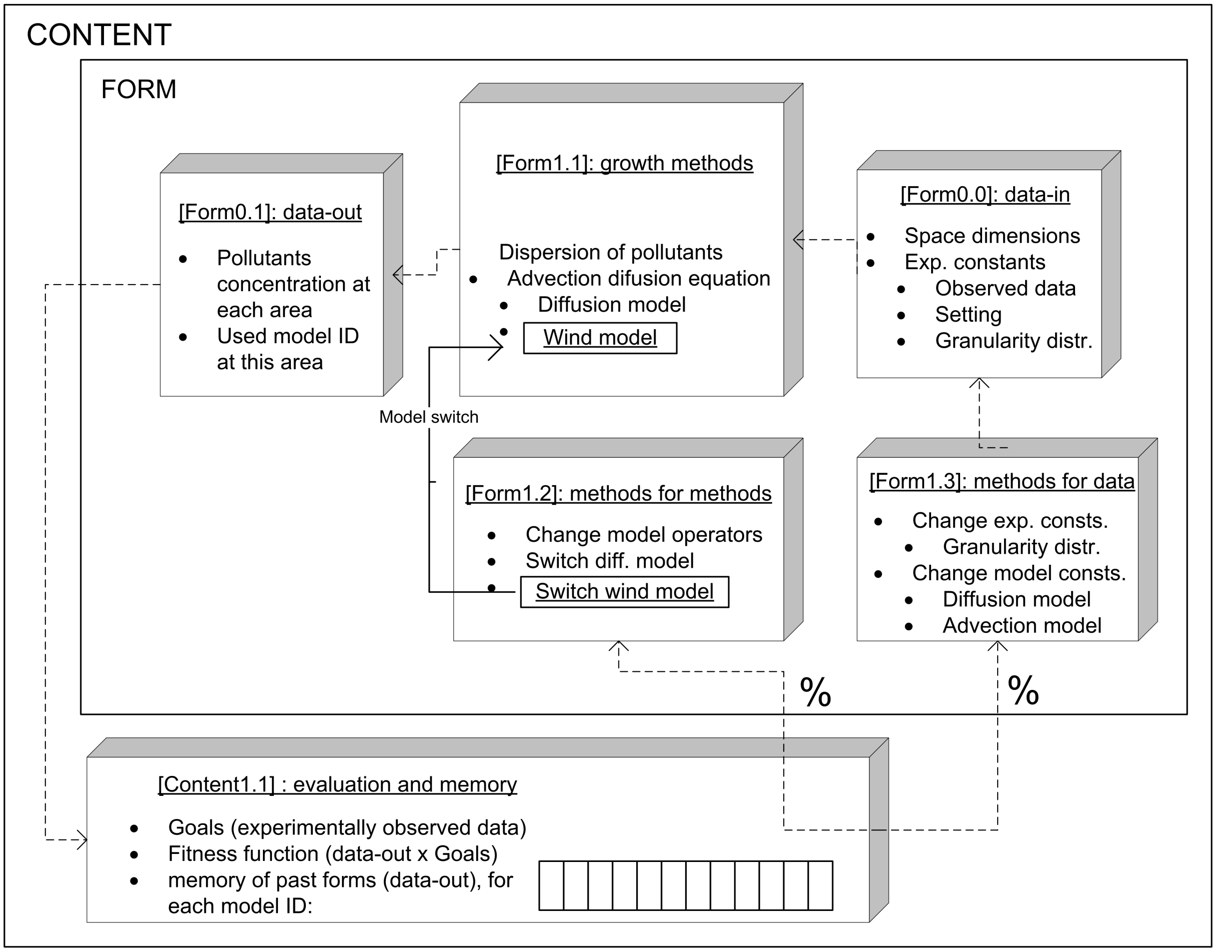

Fig. 1 Schematic diagram of self-developmental atmospheric pollutant dispersion model. Data-in, space dimensions, experiment constants and initial setting are processed by - advection diffusion equation onto data-out (pollutants concentration at processed area). Data-out is being stored (at each iteration) into the Content memory and evaluated by fitness function according to its current value, past states in memory and expected value (goals) collected from experimental measurements. Based on this evaluation specific methods from Form1.2 are applied on advection diffusion equation (Form1.1) and specific method from Form1.3 are applied on data-in (Form0.0). Thus methods (operators) and data are evolved in parallel. Both of these applications are probabilistic, denoted by "%". As more stable is the data-out Form, as less is the probability of method or data-in modification by other methods. In our first revision, only wind model evolution is being implemented by switching between several wind's model variants, see section 2.2 for variants examples.

3.3 CONTENT: Memory, Goals and Evaluation

Important part of the Content is the Memory which hold past states of the Form (data-out and method application). The Memory is updated at each iteration step. Based on the Memory, current data-out value and expected value (Goals – collected from experimental measurements) each iteration is evaluated by fitness function. According to this evaluation specific methods from Form1.2 are probabilistically applied on advection diffusion equation (Form1.1) and specific methods from Form1.3 are applied on data-in (Form0.0). Where the probability of the method correction – evolution depends on frequency of its past applications at specific setting. Generally, as more often the method at specific setting is applied, as more probable is the next application without method correction.

4 Implementation

In current state of development the model is implemented separately in Java and C++ languages. The self-development model as described in Figure 1 is implemented in Java and the part that numerically computes the model itself is implemented in C++. The reason for this separation of codes is firstly the clarity and better maintenance of the source code and secondly the speed of the numerical calculations are still much faster in C++ then in Java. The possibility to implement the numerical calculations on HW platforms such as GPU, Cell processors etc. corresponds to the second reason for code partition.

4.1 Self-developmental Model

The implementation of the model corresponds to the scheme of Figure 1. First the wind models are randomly chosen for initiation of the model. Diffusion model is keep the same throughout the whole evolution process. Each area contains just one wind model in particular time moment but it can differ in any two time moments - it depends on the developmental process that is stored into the local area memories. The depth of the memory is limited and the forgetting is simulated by the weights in each history moment. The older moment is the lower weight is set to it.

After the initiation the current form of the model is computed. The way is outlined in the next subsection. Then the new wind models have to be set to each area. Now the probabilistic selection turn comes. Based on the history of the wind models (their frequency and time distribution) and the goals that have to be achieved the best candidate is selected. It is done for all areas of the space. The evolution process continues until the specific number of iterations is reached or the evolution reaches the stable final model according to defined goals.

4.2 Numerical Model Calculation

The particular form of the model must be calculated in order to know its quality in every iteration. The model is represented by advection-diffusion equation (ADE) that is part of the general equation 1 in current implementation. The ADE is a form of partial differential equation and due to its complexity it has to be calculated by some kind of numerical scheme.

In current version of the solution the method of lines (MoL) was chosen to solve the ADE. The quality of the scheme (its precision, stability etc.) is important for the developmental process because the final quality of the evolved model depends on it. Though MoL is simple it is very universal numerical scheme which precision can be estimated and set automatically. By the universal we mean that in a case of developmental model it is changing through time and space as the different kind of wind models are choosing. In this case the numerical computation needs to set the different step sizes in different places/times. The disadvantage of MoL is its long-time computation which results in very long evolution process of the model. It can be however compensated by implementation its parallel version on some special kinds of HW platforms.

5 Conclusion and On-Going Work

In these days the designed model is being implemented in object-oriented languages as described in previous section. In our first revision, we assume the wind model evolution only by switching between several wind's model variants. In future we are going to implement the rest of designed system to obtain the whole adaptability as shown on Figure 1.

Due to the high degree of the independence among methods and data modification, our self-developmental model is ideal for parallelisation (at level of each iteration). Thus from HW point we also assume to accelerate the developmental process by using dozens of small computational units (e.g. inexpensive graphic processing units as accelerators). The computation of the numerical model can be parallelised as well. We have done some experiments with accelerating of MoL so far and achieved the speed-ups in orders of dozens to hundreds times the faster then the conventional implementations. The experiments were done on GPU graphics cards using CUDA and OpenCL frameworks.

There is also the possibility to implement more sophisticated numerical solutions of the ADE. We are now experimenting with the ELLAM [10] framework that is used to solve the ADE equations for water environment where the advection part dominates. However, it is seen from the first set of experiments that the ELLAM is suitable for atmospheric ADE as well.

6 Acknowledgements

This work is partially supported by the BUT FIT grants ”Secured, reliable and adaptive computer systems”, FIT-S-10-1, ”Information Technology in Biomedical Engineering”, GA102/09/H083 and ”Support of education of Fundamentals of Artificial Intelligence and Soft-Computing courses”, FR1613/2010/G1, and the research plan ”Security-Oriented Research in Information Technology”, MSM0021630528.

7 References

[1] R. Sheldrake. *A New Science of Life: the hypothesis of formative causation, second edition*. Los Angeles, 1985.

[2] J. Krajicek. Human inspired self-developmental model of neural network (him): Introducing content/form computing. In *The 2010 International Conference Ubiquitous Computing and Multimedia Applications*.

[3] P.J.H. Builtjes. *Air pollution modeling and its application XIV*, chapter Major twentieth century milestones in air pollution modelling and its application. Klumer Academic/Plenum Publishers, 2001.

[4] J.-S. Lin and L. M. Hildemann. A generalized mathematical scheme to analytically solve the atmospheric diffusion equation with dry deposition. *Atmospheric Environment*, 31(1):59–71, 1997.

[5] D.M. Moreira S. Wortmann, M.T. Vilhena and D. Buske. A new analytical approach to simulate the pollutant dispersion in the pbl. *Atmospheric Environment*, 39(12):2171–2178, 2005.

[6] A.G. Ulke. New turbulent parameterization for a dispersion model in the atmospheric boundary layer. *Atmospheric Environment*, 34(7):1029–1042, 2000.

[7] D.M. Moreira G.A. DEGRAZIA and M.T. Vilhena. Derivation of an eddy diffusivity depending on source distance for vertically inhomogeneous turbulence in a convective boundary layer. *Jour-*

nal of Applied Meteorology, 40(7):1233–1240, 2001.

[8] G. Kampis. *Self-Modifying systems in biology and cognitive science*. Pergamon Press, 1991.

[9] J. Koza et al. *Genetic Programming III: Darwinian Invention and Problem Solving*. Morgan Kaufmann, 1999.

[10] I. Herrera M. A. Celia, T. F. Russell and R. E. Ewing. An eulerian-lagrangian localized adjoint method for the advection-diffusion equation. *Advanced Water Resources*, 13:187–206, 1990.