DYNAMIC CLUSTERING FOR AUTO-ORGANIZED STRUCTURES IN COMPLEX FLUID FLOWS

Cyrille Bertelle, Véronique Jay, Sylvain Lerebourg
Damien Olivier and Pierrick Tranouez
Laboratoire d'Informatique du Havre
25 rue Philippe Lebon, BP 540
76058 Le Havre Cedex - France
E-mail: {Cyrille.Bertelle, Veronique.Jay, Damien.Olivier,
Pierrick.Tranouez, Sylvain.Lerebourg}@univ-lehavre.fr

KEYWORDS

agent-based simulation , fluid flow, vortex methods, auto-organization.

ABSTRACT

This paper presents an original approach for the detection of auto-organizated coherent structures in a fluid flow simulation. This method is based on vortex methods and multi-agent systems. We then use automata to simulate the interactions between these structures and the induced evolution of their stability.

INTRODUCTION

We study the simulation of fluid flows where multiscale coherent structures evolve dynamically (Tranouez et al., 2001). We focus our attention on coherent structures defined here as vortex formations. These coherent structures are for example, swirls which are formed behind obstacles. These structures are born, grow or die during the simulation under the effect of mutual or external interactions. The goal of the study presented in this paper is to implement automatic detection of such dynamical coherent structures and then to represent them in the simulation as collective entities able to evolve either through a reinforcement of their stability or devolve through destabilization and breaking-up.

In the first section, we describe the vortex methods used to compute the fluid flow in the basic level of the simulation. In the second section, we present the structure detection method. The third section describes how these detected structures are represented and how their evolution is managed. The last section gives the results of simulations made in Java with the Madkit multi-agents platform.

VORTEX BASED MODEL FOR FLUID FLOW SIMULATIONS

The basic level of the computation of the fluid flow is handled using vortex methods (Leonard, 1980) in two-dimensional space. These methods are gridless technics where virtual interacting entities, carrying vorticity, evolve following a decentralized discretization of the Navier-Stokes equations.

The fluid flow is discretized into a set of particles carrying vorticity $(\overrightarrow{X}_i, \overrightarrow{\Omega}_i)_{1\leq i\leq N}$. These particles interact among themselves and move along the flow they represent.

The particles velocity is computed with the Biot-Savart regularized formulation :

$$\overrightarrow{U}_p(\overrightarrow{X}_i,t) = \frac{1}{2\pi}\sum_{j\neq i}\overrightarrow{\Omega}_j \wedge \frac{(\overrightarrow{X}_i - \overrightarrow{X}_j)}{(\|\overrightarrow{X}_i - \overrightarrow{X}_j\|^4 + \epsilon^4)^{1/2}} \quad (1)$$

where ϵ is a small parameter which value is approximated by $h^{3/4}$, h is the minimal distance between neighbouring particles.

The particles localisation is then computed by velocities integration.

When the fluid viscosity cannot be neglected, the vortex diffusion has to be computed using the following equation :

$$\frac{d\overrightarrow{\Omega}_i}{dt} = \frac{\sum_j (V_i\overrightarrow{\Omega}_j - V_j\overrightarrow{\Omega}_i)\exp\left(-\frac{\|\overrightarrow{X}_i-\overrightarrow{X}_j\|^2}{4\nu\Delta T}\right)}{4\pi\nu(\Delta T)^2} \quad (2)$$

Proceedings 14th European Simulation Symposium
A. Verbraeck, W. Krug, eds.

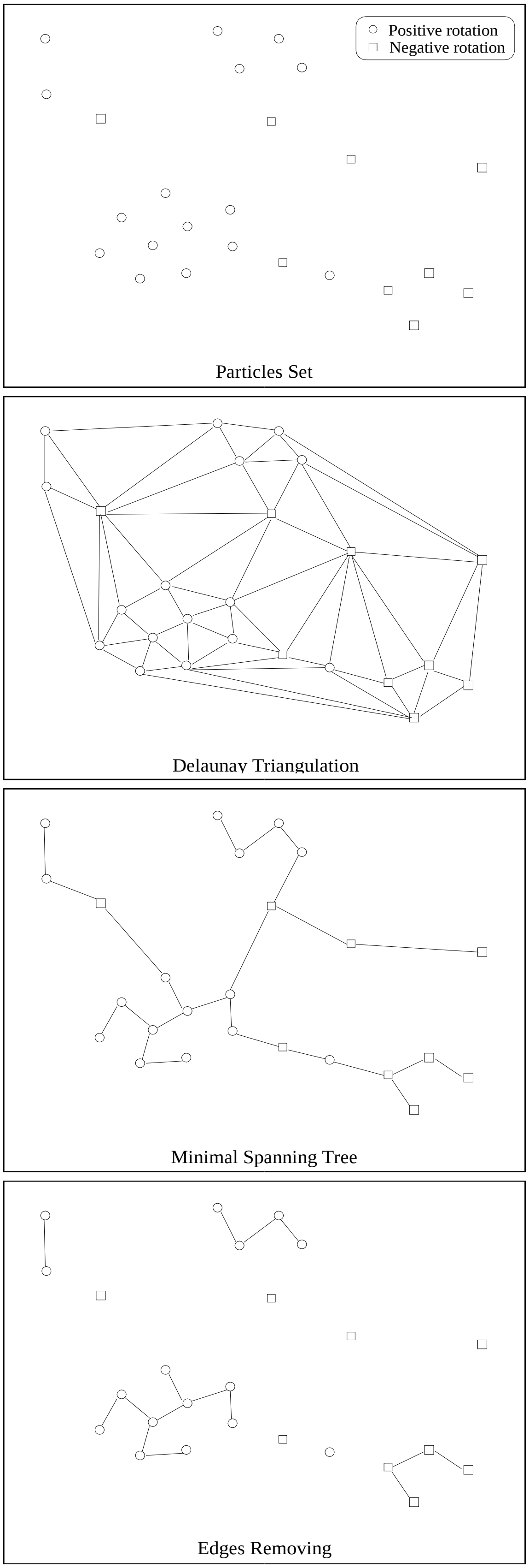


Figure 1: Successive sub-steps for clusters formation

COHERENT STRUCTURES DETECTION IN FLUID

The coherent structures detection is split in two main steps. First we detect clusters of close particles sharing certain properties. Then for each cluster, we compute the ellipse closest to its border, as ellipses are the usual shapes of natural swirls.

The fisrt step is described by the figure 1. The Delaunay triangulation is performed over the whole set of particles. Afterward, the associated minimal spanning tree is computed. In this tree, we break the relatively too long edges and the edges connecting contrarotative particles, i.e. of different rotation sense. At the end of this step, many small trees are obtained and represent emergent clusters formations.

The convex hull of each cluster is then computed, using Graham algorithm (Graham, 1972). Finally, an ellipse based identification is made from these convex hulls. This identification consists in the minimization of the square of the algebraic distance between the ellipse and the convex hull:

$$D(A) = \sum_{i=1}^{N} F(A, X_i)^2$$

where

$$F(A, X) = ax^2 + bxy + cy^2 + dx + ey + f = 0;$$

is the ellipse algebraic equation, and $X_i = (x_i, y_i)$ is the co-ordinates of one of the N points of a convex hull.

This minimization problem lacks one constraint to admit an unique solution. We therefore add a condition which can be, for example, one of the following :

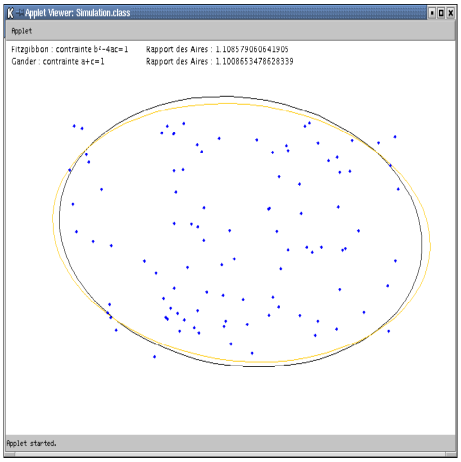
- Gander criterion (Gander et al., 1994):
  $a + c = 1$
- Fitzgibbon criterion (Fitzgibbon et al., 1996):
  $4ac - b^2 = 1$

An exemple of such identifications with these two conditions is shown in the figure 2 where the darker ellipse corresponds to Fitzgibbon criterion and the lighter ellipse corresponds to Gander criterion.

This identification is completed with the use of compacity criteria which increase the point density on the convex hulls to obtain ellipses whose frontier is nearest to the whole set of particles in the cluster. Finally the ellipse eccentricity is computed for each identification to remove the too flat ones. Indeed, natural swirls optimal shape is the circle.

AUTOMATA BASED MODELIZATION FOR FLOW STRUCTURES EVOLUTION AND STABILIZATION

Once the fluid structures detected, we have to model their evolution during the simulation. Different types of



```
Fitzgibbon : contrainte b²-4ac=1      Rapport des Aires : 1.108579060641905
Gander : contrainte a+c=1             Rapport des Aires : 1.1008653478628339
```

Figure 2: Ellipse identification on particles cluster

interactions have to be taken into account. Contrarotative close structures destabilize one another. Isorotative close structures increase one another stability. We model these qualitative evolutions in each structure with an eco-agent (Drogoul and Dubreuil, 1992) built on an automaton with multiplicities (Bertelle et al., 2001). This kind of automaton is well suited to represent interacting agent behaviour in multi-agent system for solving classical problem in artificial intelligence in a distributed way. This resolution is called eco-resolution. In the distributed description of the fluid flow studied in this paper, the dynamic structures manage their mutual interactions through an eco-resolution which leads to solve the non linear aspects of the whole system of interactions.

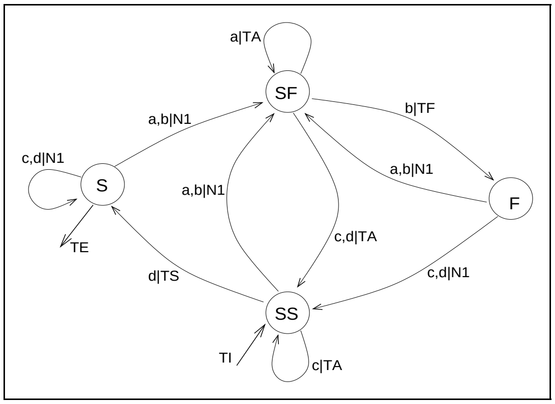


Figure 3: Eco-agent behaviour as automaton with multiplicities (in transducer form)

An eco-agent is characterized in its generic formulation by:

- an internal state which is one of the following: to be satisfied (S), to search satisfaction (SS), to flee (F) or to seek to flee (SF); the initial state is (SS) and the final state is (S); the final state of satisfaction corresponds to the goal of the agent;
- elementary perceptions which are:
  - to be attacked (by other agents): event denoted A;
  - to perceive some intruder (such as other agents preventing it to be satisfied): event denoted I.
- elementary actions which are:
  - to flee (TF);
  - to satisfy itself (TS);
  - to attack other agents (TA);
  - to do nothing ($N1$).

The behaviour of the eco-agent can be represented by the automaton of the figure 3. The states are those described before and the transitions are labelled with couples constituted by the corresponding perceptions and actions (separated by a vertical line). These actions are those described before and the perceptions are all the four combinations of existence or non existence of the two elementary perceptions previously described : $a = (A, I)$, $b = (A, \overline{I})$, $c = (\overline{A}, I)$, $d = (\overline{A}, \overline{I})$.

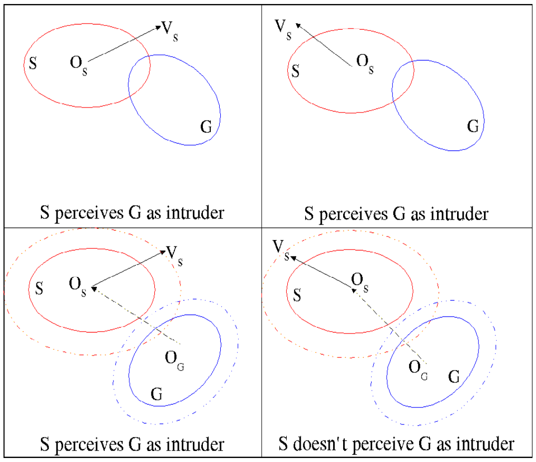


Figure 4: Intruders perceptions

Using eco-agents as behaviour model for a specific problem consists in fact in defining the two elementary perceptions (to be attacked and to perceive intruders) and the four elementary actions (to do nothing, to attack, to flee, to satisfy himself). That's what we define now for our specific problem, fluid flow structures interactions:

- Perceiving intruders means a structure intercepts another one or at least is on a collision course with it. The nearby structures are detected by adding an additional interaction zone around each structure. The figure 4 sums up the different possibility for contrarotative structures in a close vicinity and the associated perceptions.

- Attacking a structure means sending the corresponding message to this one.
- Being attacked means being on the receiving end of said message.
- Fleeing means acting upon one's destabilization. In this case, the structure reduces its dimension and generate elementary particles on its previous frontier to preserve the whole rotational (see figure 5).

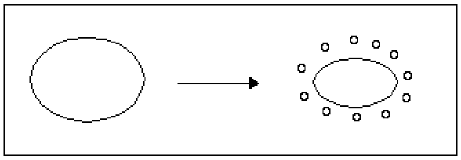


Figure 5: Structure flight as reducing its dimension

- Satisfying oneself means increasing one's stability. In this case, the structure tries to aggregate nearby particles of the same rotational. The aggregation process consists in the research of the minimal spanning tree from its frontier, linking particles of the same rotational that the structure itself. If the Delaunay triangulation associated to this minimal spanning tree is compact (that is, if the ratio of the sum of all triangles area over its convex hull area, is near the numeric value 1.0), the particles are aggregated to the structure and a new ellipse identification is then made with them. The figure 6 shows such process: the left particles set will be aggregated to the structure. The top particles set will not be aggregated because its associated triangulation is not compact, probably due to the fact that it may be the formation of a new vortex.

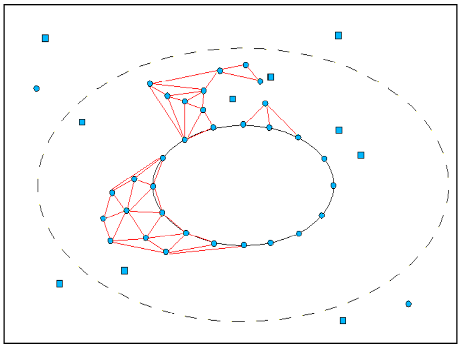


Figure 6: Aggregation from structure in satisfaction process

IMPLEMENTATION AND SIMULATION RESULTS

An implementation of a simulation using this model for fluid flow has been carried out in Java (Lerebourg, 2002). The structures interaction is implemented using agent programming based on the Madkit platform (Gutknecht and Ferber, 1997). Qualitative graphical results show realistic simulations for the stabilization/destabilization process of interacting structures after their automatic detection. The figure 7 shows the evolution of a ring of fluid structures. This ring is initialy constituted by:

- a large structure of negative rotation (lighter line) in the center of the ring;
- two cercles of structures of positive rotation (darker line) around the previous one; the inner one contains larger structures and the outer one contains smaller structures;
- a sequence of contrarotative particles couples, inducing a displacement from left to right and which are going to enfold the ring;
- a constant linear flowing is created from left to right in the domain.

This experimentation shows the effect of the rotation of the ring on itself. The elementary particles play a destabilization role. After some step (3rd picture) the central contra-rotative structure begins to reduce itself and is finally destroyed (4th picture), generating elementary particles diffusing in the flow.

CONCLUSION

The study presented here is the first step of the implementation of a multi-level fluid flow description. Automatic process manage interactions between emergent structures and between the two scale levels implemented (macro-micro interactions). The usual way of modelling fluid flows uses meshes. It is a global method. Therefore, it handles poorly local discontinuities. Our approach is local and it allows to model more heterogeneous complex systems where entities of other nature will be represented (like obstacles or biological ones such as bacteries). The local approach of the model based on autonomous entities (as per the agent paradigm), unable us to describe accurately inter-scales phenomena, carried by the fluid flow in aquatic environments.

REFERENCES

Bertelle, C., Flouret, M., Jay, V., Olivier, D., and Ponty, J.-L. (2001). Automata with multiplicities as behaviour model in multi-agent simulations. In *SCI'2001*, Orlando (USA).

Drogoul, A. and Dubreuil, C. (1992). Eco-problem-solving: results of the n-puzzle. In Demazeau, Y. and Werner, E., editors, *Decentralized Artificial Intelligence III*, pages 283–295. North Holland.

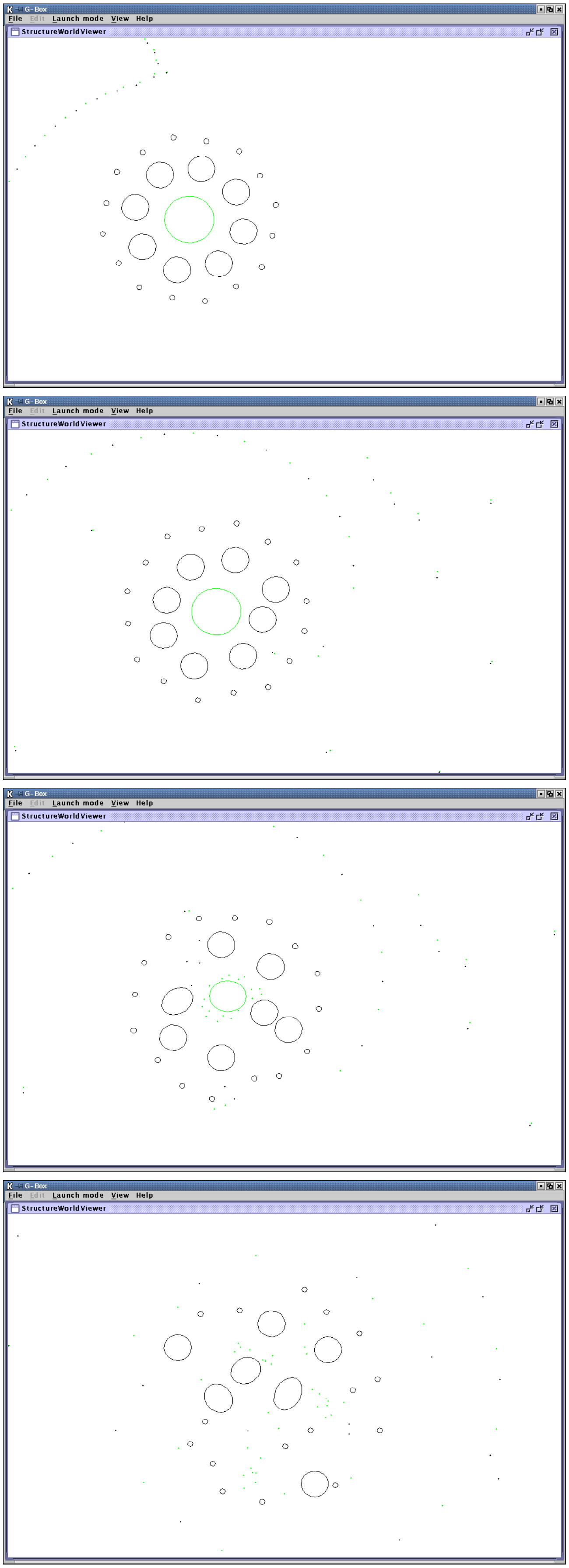


Figure 7: Evolution of a ring of structures

Fitzgibbon, A., Pilu, M., and Fisher, R. (1996). Direct least squares fitting of ellipses. Technical report, Edinbourg University.

Gander, W., Golub, G., and Strebel, R. (1994). Fitting of circles and ellipses - least squares solution. Technical report, ETH Zurich.

Graham, R. (1972). An efficient algorithm for determining the convex hull of a finite planar set. *Information Processing Letters*, 1:132–133.

Gutknecht, O. and Ferber, J. (1997). Madkit: Organizing heterogeneity with groups in a platform for multiple multi-agents systems. Technical report, LIRMM, Montpellier University, http://www.madkit.org.

Leonard, A. (1980). Vortex methods for flow simulation. *Journal of Computational Physics*, 37:289–335.

Lerebourg, S. (2002). Clustering dynamique appliqué aux écoulements fluides complexes. Technical report, Le Havre University.

Tranouez, P., Bertelle, C., and Olivier, D. (2001). Changing the level of description of a fluid flow in a agent-based simulation. In *ESS 2001 Conference*, Marseilles (France).