Project Proposal

Title: ALICE

Subtitle: Geometry and Light

Scientific leader: Bruno Lévy

Proposed INRIA theme: Cog-D

INRIA scientific and technological challenges:[1]
Combining simulation, visualization and interaction

Keywords:
Computer Graphics, Geometry Processing, Light Simulation

[1]The seven INRIA scientific and technological challenges are:

1. Designing and mastering the future network and communication services infrastructures
2. Developing multimedia data and information processing
3. Guaranteeing the reliability and security of software-prevalent systems
4. Coupling models and data to simulate and control complex systems
5. Combining simulation, visualization and interaction
6. Modeling living beings
7. Fully integrating ICST into medical technology

Contents

1 Project team composition

Permanent members

Laurent Alonso	CR Inria
Xavier Cavin	CR Inria
Bruno Lévy	CR Inria, Scientific leader

Post-doctoral fellows

Nicolas Ray	INRIA
Bin Wang	ACI Grid

PhD students

Laurent Castanié	INPL/Earth Decision Sciences
Wan Chiu Li	INRIA
Grégory Lecot	INRIA/VSP Technology
Rodrigo Toledo	INRIA
Bruno Vallet	Polytechnique

Software developers

Jonathan Muller	Associate Engineer
Christophe Mion	Associate Engineer

Administration

Isabelle Herlich	TR Inria

External collaborators

Jean-Claude Paul	DR Inria - Tsinghua University

2 Overall objectives

This document describes our proposal for an Inria Project on Computer Graphics. The fundamental aspects of this domain concerns the interaction of *light* with the *geometry* of the objects. The lighting problem consists in designing accurate and efficient *numerical simulation* methods for the light transport equation. The geometrical problem consists in developing new solutions to *transform and optimize geometric representations*. Our original approach to both issues is to restate the problems in terms of *numerical optimization*. We try to develop solutions that are *provably correct*, *scalable* and *numerically stable*.

◇ By provably correct, we mean that some properties/invariants of the initial object need to be preserved by our solutions.

◇ By scalable, we mean that our solutions need to be applicable to data sets of industrial size.

◇ By numerically stable, we mean that our solutions need to be resistant to the degeneracies often encountered in industrial data sets.

To reach these goals, our approach consists in transforming the physical or geometric problem into a numerical optimization problem, studying the properties of the objective function and designing efficient minimization algorithms.

Besides Computer Graphics, we have cooperations with researchers and people from the industry, who experiment applications of our general solutions to various domains, comprising CAD, industrial design, oil exploration, plasma physics...

3 Historic context and state of the art

3.1 Geometry in the ISA project

ALICE is one of the three INRIA project proposed by former ISA members. This section summarizes the scientific evolution of the research groups within ISA that yielded those three project proposals. More specifically, since VEGAS and ALICE both do research in geometry, this section explains the two different visions of geometry developed by these two project proposals.

One of the principal research orientations of the ISA project was computer vision and augmented reality. A clearly identified group headed by Marie-Odile Berger developed this approach, so it was natural for them to propose the creation of the MAGRITTE project.

The other principal research orientation of the ISA project was physically-based light simulation. The main challenges of this domain are both geometrical (visibility complex, surface intersections) and computational (numerical resolution of an integral equation). To deal with the geometrical problems, a 'geometry research group' was created within ISA. The missions of this group were the following three ones:

1. To generate the surfacic geometry of the scene from volumic Constructive Solid Geometry descriptions, design new *intersection algorithms*. More precisely, given the equation of two surfaces, the goal is to obtain a parameterization of the intersection;

2. Find ways of *attaching photometric properties to the geometric objects* in the scene. For the numerical simulation of light (i.e., energy transfers), it is necessary to find parameterizations with a constant Jacobian (i.e., energy-preserving parameterizations);

3. Optimize point-to-point *visibility* requests (they are massively issued by light simulation algorithms).

To make the scope of this research as general as possible, the two main classes of surfacic representations were considered, i.e., algebraic surfaces and meshed models. When using algebraic surfaces, to represent complex objects, piecewise defined surfaces are used. The geometric continuity between the charts is the main challenge of the Geometric Design domain of research. In other words, Geometric Design is concerned with G^k continuity (Geometric Continuity), defined as follows: a surface of class G^k is a surface for which a parameterization of class C^k exists. To construct a G^1-continuous object with a set of parametric polynomial surfaces defined over triangles, it was proved that at least degree 4 is required[Her87]. We considered the main class of algebraic surfaces used to represent geometry in the Geometric Design community, i.e., rational fractions called NURBS (Non-Uniform Rational B-Splines). However, it quickly appeared to us that the solution of the mathematical problems expressed with NURBS (surface intersection and energy-preserving parameterization) do not have a closed form in general. As a consequence, two complementary approaches were developed in parallel by ISA:

◇ The first approach considered an exact solution of a simplified version of the problems, and limited the study to polynomial surfaces of degree 2 (i.e., quadrics). The main advantage is that closed forms for both the problem of surface intersection (point 1 of the research program on the previous page) and energy-preserving parameterization (point 2) could be derived. Using Galois's group theory and projective geometry, it was possible to derive a general expression of the intersection of two quadrics, with a provable minimum number of square roots. This elegant theoretical result answered several open questions and was welcomed by the Computational Geometry community. As a matter of fact, the projective geometry background acquired by the group was also successfully applied to start the study of the visibility complex (point 3). The VEGAS project headed by Sylvain Lazard and Sylvain Petitjean continues to develop this approach;

◇ the second approach kept the initial specification of the problem: the geometry is represented by a set of high-order surfaces or by meshed models. The group already knows that no closed form can be derived, for this reason, we developed approximated solution mechanisms, based on applied mathematics and numerical analysis. Being able to process industrial-scale models (with millions of primitives) was also one of the major preoccupations of the group. Applied to mesh models, the solutions we developed had different applications in texture mapping (point 2 of the research program). More generally, our solutions were welcomed by the Digital Geometry Processing community, a new discipline of Computer Graphics that recently emerged. In addition, we also started to apply our geometry processing tools to the numerical simulation of light. These two aspects - Digital Geometry Processing and the numerical simulation of light - are the directions of research proposed by the ALICE project, described in this document.

The next sections introduces the Computer Graphics domain, summarizes its major historical evolutions, outlines the emergence of the Digital Geometry Processing discipline, and describes our strategy. Short term research projects in Digital Geometry Processing and Light Simulation are detailed later (Sections 6.1 and 6.2 respectively). Longer term research projects are outlined in Section 6.3.

[Her87] G.J. Herron. *Techniques for Visual Continuity*, pages 163–174. G. Farin ed. SIAM, 1987.

A B C

Figure 1: *Computer Graphics in the 70's: Henri Gouraud's shading algorithm. His algorithm is still used today, 30 years after. On the left: to obtain the data, Sylvie Gouraud accepted to be manually digitalized (3D scanner did not exist at that time).*

3.2 Computer Graphics

Computer Graphics is a quickly evolving domain of research. It started in the early 70's. An interest group emerged in the University of Utah. At the beginning, the main problem was to design algorithmic and hardware solutions to display 3D geometry on a computer screen, and classic algorithms were invented in this period. Ivan Sutherland and David Evans developed hidden surface removal algorithms, motivated by a flight simulator application. The company they created still exists today (see `http://www.es.com`). To improve the appearance of computer generated images (see Figure 1-B), Henri Gouraud developed his famous "smooth shading" algorithm (Figure 1-C). More than 30 years after, his algorithm is still in use in all 3D computer graphics hardware and software.

Today, the whole picture has completely changed. Computer Graphics is now a vast domain of research, that has exploded into many different disciplines. To name but a few, the main disciplines are:

- ◇ Image synthesis and light simulation
- ◇ Shape modeling, Digital Geometry Processing
- ◇ Data acquisition, reconstruction
- ◇ Animation and physically based modeling
- ◇ Computer Vision and Image Processing
- ◇ Texture mapping and texture synthesis

In parallel with the evolution of the scientific community, computer graphics hardware have made huge advances, evolving faster than Moore's law prediction[2]. In addition, computer graphics hardware has became programmable, which offers new ways of generating computer images.

ALICE's general goal, as an INRIA research project, is to reach the two objectives of the INRIA strategic plan in Computer Graphics, i.e., both *pushing forward the scientific state of the art* and doing *technology transfers*. Considering these two objectives, the evolution trends of Computer Graphics, i.e., explosion in a multitude of disciplines and tight coupling with a quickly evolving technology, cause the following difficulties:

- ◇ How can we invent a research program likely to impact the largest possible part of the Computer Graphics community ?
- ◇ How can we make sure that the impact of our research results will not be reduced - or even made obsolete - by technology evolutions ?

[2]The highly parallel nature of computer graphics makes it possible to double performances by just adding a processing unit on the graphic board. This has been done extensively by the industry those last few years

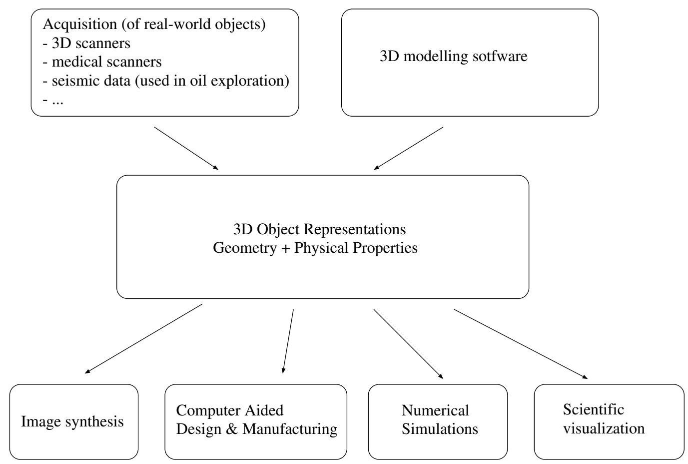
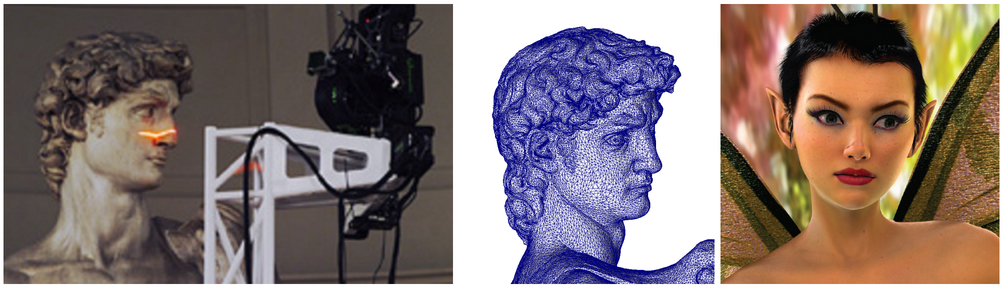


Figure 2: *Geometric representations are central in Computer Graphics. Advances in this discipline may have significant impact both upstream (data acquisition, data modeling) and downstream (image synthesis, CAD/CAM, simulations and visualization).*

To overcome these two difficulties, our strategy is to analyze what is done in general in Computer Graphics, what is the general trend in the evolution of the technology, and try to identify the most general scientific problems. We show later in this document our strategy/proposals to solve these problems.

Computer Graphics disciplines: identify general problems As shown in Figure 2, the common starting point of all Computer Graphics disciplines is the creation of 3D data. This 3D data can be either acquired from real objects, or created ex-nihilo by using 3D modeling software. Downstream, different classes of applications can use this data. In this picture, the *representation of the 3D objects* clearly appears to play a central role. In addition, these 3D objects need to be enriched with *physical properties* attached to them. For instance, in image synthesis, these physical properties correspond to the photometric characteristics of the materials. In numerical automobile crash tests, they can correspond to stress and strain… As a consequence, we think that advances in object/properties representation are likely to impact both upstream and downstream disciplines. More specifically, we try to answer the following question: given an initial geometric representation of an object, given the constraints of a discipline, how can we transform it to satisfy the constraints ?

Computer Graphics technology: identify the general trend These last 10 years, Computer Graphics hardware started a fast paced evolution. Every six months, the set of available features exposed by the hardware is enriched with a new set of functionalities. This continuously changes the way computer images can be generated. However, general concepts do not change over years. In general, a computer graphics pipeline processes a stream of vertices. Those vertices are assembled into primitives. Those primitives are rasterized onto the screen, which produces a stream of pixels. Additional data can be introduced into those streams, in the form of multi-dimensional arrays called textures. Today's hardware still complies with this general specification, introduced 30 years ago. In order to preserve the impact of our research from the evolutions of this hardware, our strategy is to consider problems positioned upstream relative to hardware evolution, and to consider only the stable part of the hardware design. The general formulation of these problems can be stated as follows: given a representation of a geometric object and its associated physical properties, how can we transform it to make it efficiently displayed by the hardware? Clearly, this is just an instance of the problem described in the previous paragraph. The next section shows how both problems have evolved these last 30 years, and we present later in this document a research strategy that is likely to answer both questions.



A B C

Figure 3: *Computer Graphics in the 2000's. Compare this Figure with Figure 1: huge advances were made. However, the basic problems still remain unsolved, i.e., finding common representations for data acquisition (A), modeling (B) and image generation (C).*

3.3 Computer Graphics in the 2000's

These last few years, advances in both data acquisition, hardware and simulation make it possible to achieve a high degree of realism. This realism is obtained by using highly detailed geometry and highly detailed physically-based models (see Figure 3):

- ◇ Early 3D data sets shared by the community (the famous Utah teapot, Sylvie Gouraud's digital face shown in Figure 1-A . . .) were all constructed by hand. The "3D fax machine" project, developed by Marc Levoy, has made the notion of 3D range laser scanners popular. The Bunny and Buddha data sets, ubiquitously used by the community, have shown the feasibility of this data acquisition process. 3D scanners can now automatically digitilize large objects (Figure 3-A), and this replaces tedious manual interventions (Figure 1-A). Highly detailed meshes can be automatically produced from complex objects, including whole human bodies and mechanical parts. Figure 3-B shows a scanned mesh of the David statue, part of the Digital Michelangelo collection;

- ◇ elaborate lighting models now replace the simple diffuse Lambert law implemented in early systems. These models now take complex light-matter interaction schemes. For instance, subsurface scattering takes into account the way light is scattered in translucent materials, such as human skin. Figure 3-C shows this lighting model computed in real-time by modern computer graphics hardware.

As can be seen, huge advances have been accomplished since the early ages. The quality of today's computer generated images has so much progressed that synthetic images can be less and less distinguished from photos. In other words, at first sight, it seems that most Computer Graphics problems have been solved. However, several important issues remain open. For instance, a 3D scanner produces an unstructured set of points by regularly sampling a 3D object. From this set of points, reconstruction methods can produce a 3D mesh. However, the so-constructed mesh (Figure 3-B) is difficult to use in applications. A scanned mesh is composed of a large number of elements (10 million triangles is an average size), this makes it difficult to efficiently display and modify these meshes. Applying numerical simulations is even more difficult, due to both performance and numerical stability problems. The same difficulties occur when using objects constructed by a 3D modeling software. In other words, the bottleneck of the process was not overcame, it was simply pushed from the data acquisition to the data processing step. To study this class of problems, "Digital Geometry Processing", a new discipline, has recently emerged. The next section introduces this discipline. ALICE's strategy in this discipline will be presented in Section 6.1.

4 Scientific foundations

4.1 Digital Geometry Processing

Digital Geometry Processing recently appeared (in the middle of the 90's) as a promising avenue to solve the geometric modeling problems encountered when manipulating meshes composed of million elements. Since a mesh may be considered to be a *sampling* of a surface - in other words a *signal* - the *digital signal processing* formalism was a natural theoretic background for this discipline (see e.g. [Tau95]). The discipline then studied different aspects of this formalism applied to geometric modeling. We quickly introduce below the main aspects of Digital Geometry Processing, namely multiresolution, discrete fairing and mesh parameterization

Multiresolution: Together with the sampling aspects, resolution problems naturally appeared: to accurately sample a signal, Shannon-Nyquist's law states that the sampling frequency needs to be at least equal to twice the highest frequency of the signal. For a signal that has a wide spectrum of frequencies, this is a waste of both storage space and computation time. Multiresolution methods provide a natural answer to this problem, by splitting the signal into a set of components (named harmonics), with the possibility of using a sampling rate adapted to each harmonic. Peter Schroeder was one of the pioneers who applied this formalism (in particular, a family of functions called wavelets) to global light simulation [Sch94] and Steven Gortler applied them to geometric modeling problems. However, constructing a multiresolution representation from a given geometric object is a non-trivial problem. Motivated by data simplification and Level-of-Detail-based visualization, Hugues Hoppe developed the *Progressive Mesh* data structure in [Hop96]. A progressive mesh is represented by a simplified base mesh and a series of refinement operations. Transforming an object into a progressive mesh naturally separates the different geometric frequencies of the initial object. Based on this observation, progressive meshes (or similar data structures) were used to construct multiresolution geometric representations [EDD+95] and to define multiresolution modeling and processing tools acting on them [KCVS98], [GSS99].

Discrete fairing: Adapting to meshed models all the modeling tools available with the "Curves and Surfaces" representation is another challenge of the Digital Geometry Processing discipline. In "Curves and Surfaces" representations, the geometry is represented by a set of parametric surfaces. Time and effort has been devoted to the problem of optimizing the shape of a surface, by minimizing a "fairness" criterion. Fairness is often defined using notions from differential geometry (mean curvature, Gaussian curvature ...) or approximation of physics (thin-plate energy). In general, optimizing the fairness means solving a Partial Differential Equation [BW90]. Adapting this formalism to the case of a discrete mesh model was an active research area. Kobbelt coined the term *discrete fairing* in [Kob97] to qualify this family of approaches.

[Tau95] G. Taubin. A signal processing approach to fair surface design. In *SIGGRAPH Conference Proceedings*, pages 351–358. ACM, 1995.

[Sch94] P. Schroeder. *Wavelet Methods for Global Illumination*. PhD thesis, Princeton University, 1994.

[Hop96] H. Hoppe. Progressive meshes. In *SIGGRAPH Conf. Proc.*, pages 99–108. ACM, 1996.

[EDD+95] M. Eck, T. DeRose, T. Duchamp, H. Hoppe, M. Lounsbery, and W. Stuetzle. Multiresolution analysis of arbitrary meshes. In *SIGGRAPH Conference Proceedings*, pages 173–182. ACM, 1995.

[KCVS98] L. Kobbelt, S. Campagna, J. Vorsatz, and H.P. Seidel. Interactive multi-resolution modeling on arbitrary meshes. In *SIGGRAPH Conference Proceedings*, pages 105–114, 1998.

[GSS99] I. Guskov, W. Sweldens, and P. Schröder. Multiresolution signal processing for meshes. In *SIGGRAPH Conference Proceedings*, pages 325–334. ACM, 1999.

[BW90] M.I.G. Bloor and M.J. Wilson. Using PDEs to generate free-form surfaces. *CAD*, 22, 1990.

[Kob97] L. Kobbelt. Discrete fairing. In *Proceedings of the Seventh IMA Conference on the Mathematics of Surfaces*, pages 101–131, 1997.

Mesh parameterization: Mesh parameterization is another problem for which the Digital Geometry Processing have been investing much activity these last few years. "Curves and Surfaces" geometric models are represented by parametric functions. This representation is useful for many application domains, including attaching properties to the surface (they can be represented by 2D data structures in parameter-space), or meshing algorithms. For this reason, methods to obtain a parametric representation from a mesh model were investigated. In his pioneering work[Flo97], motivated by a Spline fitting problem, Michael Floater had the idea to use Tutte's barycentric mapping theorem[Tut60] to construct a piecewise linear parameterization of a triangulated mesh homeomorphic to a disc. Many papers have been then published on this specific topic, relaxing the constraint of using a fixed convex boundary in parameter space, and minimizing different deformation criteria, adapted to different application domains. The SIGGRAPH conference now devotes an entire session to this specific issue, and Michael Floater reports more than 20 parameterization papers published each year in major conferences and journals. His recent survey[FH04] lists the most significant advances in this area.

Open problems: Although many advances have been made in the Digital Geometry Processing area, important problems still remain open. As mentioned in Section 3.3, even if shape acquisition and filtering is much easier than 30 years ago, a scanned mesh composed of 30 millions of triangles cannot be used directly in real-time visualization or complex numerical simulation. For this reason, automatic methods to convert those large meshes into higher level representations are necessary. However, these automatic methods do not exist yet. For instance, the pioneer Henri Gouraud (see Figure 1) often mentions in his talk that the *data acquisition* problem is still open. Malcolm Sabin, another pioneer of the "Curves and Surfaces" and "Subdivision" approaches, mentioned during several conferences of the domain that constructing the optimum control-mesh of a subdivision surface so as to approximate a given surface is still an open problem. More generally, converting a mesh model into a higher level representation, consisting of a set of equations, is a difficult problem for which no satisfying solution have been proposed. This is one of the long-term goals of international initiatives, such as the `AIM@Shape` European network of excellence.

Motivated by the two principal application domains of ISA (oil and gas exploration, and industrial design / visualization), we contributed to the Digital Geometry Processing discipline at the early stages of its development. Our research obtained so far, our strategy and positioning, and our plans for future research are presented in Section 6.1. The Digital Geometry Processing solutions developed by the ALICE project aim at constructing from an initial object a geometric representation taking into account the constraints of numerical simulation. Since we are especially motivated by the Computer Graphics domain of research, the simulation of light is a natural candidate for the numerical simulation processes that we want to experiment. For this reason, we plan to continue the research direction in light simulation started by the ISA project. The next section presents this discipline, its problems, and the solutions developed by the community.

[Flo97] M. Floater. Parametrization and smooth approximation of surface triangulations. *Computer Aided Geometric Design*, 14(3):231–250, April 1997.

[Tut60] W. Tutte. Convex representation of graphs. In *Proc. London Math. Soc.*, volume 10, 1960.

[FH04] M. S. Floater and K. Hormann. Surface parameterization: a tutorial and survey. In M. S. Floater N. Dodgson and M. Sabin, editors, *Advances on Multiresolution in Geometric Modelling*. Springer-Verlag, 2004.

4.2 Numerical simulation of light

Numerical simulation of light means solving for light intensity in the “Rendering Equation”, an integral equation modeling energy transfers (or light *intensity* transfers). The Rendering Equation was first formalized by Kajiya[Kaj86], and is given by:

$$I(x,x') = g(x,x')\left[\varepsilon(x,x') + \int_S \rho(x,x',x'')I(x',x'')dx''\right]$$

where:

$I(x,x')$	denotes the intensity of light passing from point x' to point x	
$g(x,x')$	is a “geometric” term (depends on the distance between x and x', on the relative direction of their normals, and on the visibility between x and x')	(1)
$\varepsilon(x,x')$	denotes the intensity of emitted light from x' to x	
$\rho(x,x',x'')$	denotes the intensity of light scattered from the direction of x'' to the direction of x at point x'	

Computing global illumination (i.e., solving for intensity in Equation 1) in general environments is a challenging task. Global illumination may be considered in terms of computing the interactions between the *lighting signal* and the *geometric signal* (i.e., the scene). These interactions occur at various *scales*. This issue belongs to the same class of problems encountered by Digital Geometry Processing, described in the previous section. As a consequence, the *signal processing* family of approaches is again a well-suited formalism. As such, the *multi-scale* approach is a natural choice, which dramatically improves performances. When computing an energy transfer, the main idea consists in considering the problem at a suitable scale. For large-scale variations of lighting, grouping primitives makes computations more efficient in uniform zones. For small-scale variations, adaptively subdividing primitives enables capturing lighting variations of high frequency. The same approach is used by ray-tracing methods, with the difference that the adaptive discretization is performed in ray-space rather than in primitive-space, based on the theory of sampling.

Environments composed of a large number of primitives, such as highly tessellated models, show a high variability of these scales. The following four classes of interactions can be considered :

- ◇ **large scale** → **large scale**: uniform lighting conditions applied to simple geometry;
- ◇ **large scale** → **small scale**: uniform lighting conditions applied to complex geometry;
- ◇ **small scale** → **large scale**: irregular lighting conditions applied to simple geometry. For instance, when a complex object casts shadows onto a simple one;
- ◇ **small scale** → **small scale**: irregular lighting conditions applied to complex geometry. For instance, when a complex object casts shadows onto itself.

In addition, these methods are challenged with more and more complex materials that need to be taken into account in the simulation. The simple diffuse Lambert law has been replaced with much more complex reflection models. The goal is to create synthetic images that no longer have a synthetic aspect, in particular when human characters are considered.

Historically, even before the Rendering Equation was formalized by Kajiya in 1986, the “Ray-Tracing” approach was proposed by Whitted[Whi80] in 1980. To deal with both the geometric complexity and the physical complexity of the data, more elaborate computation strategies have been developed, namely hierarchical finite element methods, fast multipole

---

[Kaj86] J. Kajiya. The rendering equation. In *Computer Graphics (Siggraph)*, 1986.

[Whi80] Turner Whitted. An improved illumination model for shaded display. *Communications of the ACM*, 23(6):343–349, June 1980.

methods and monte-carlo integration. The next sections give a quick overview of them. Section 6.2 will detail our strategy and directions of research.

Finite Element Modeling and Wavelets

Historically, to improve the results obtained by early methods (ray-tracing, bi-directional path-tracing), Finite Element Methods (FEM) were first used to numerically solve Equation 1 (see e.g. [CW93]). Some simplifying assumptions were made, considering that all the materials scatter light in an isotropic manner. In other words, the function ρ of the Rendering Equation (also called a BRDF for Bidirectional Reflectance Distribution Function) was considered to be a constant. Under this assumption, the intensity of light emitted from a given point of the scene does no longer depend on the direction. In all the computation, the direction-dependent intensity (called radiance) is replaced with a direction-independent intensity (called radiosity). Removing the directional degree of freedom dramatically simplifies the equation and the computation process. However, computing and representing the radiosity for a complex model still remained a difficult problem. The FEM method applied to radiosity simulation was formalized in [GTGB84]. To speed-up computations, one of the main important improvement to FEM radiosity was the introduction of hierarchical methods [HSA91]. Wavelets then appeared as a natural and efficient formalism to study those hierarchical methods [GSCH93],[Sch94]. Using these hierarchical function bases gives a stronger theoretical background to adaptive subdivision. Simultaneously, the Galerkin method [Zat93] was introduced in the formalism. Using this formalism, the rendering equation is projected onto a basis of orthogonal functions. The problem is then restated as a linear equation. In the case of hierarchical radiosity, the function basis is a hierarchically refined wavelet basis. As can be seen, the evolution of radiosity methods is a constant oscillation between intuition and formalization. The global trend is to formalize the problem and restate it in terms of abstract entities. This unleashes the power of formal analytical methods. For instance, functional analysis was used in [Arv95] to analyze the error bounds. As shown further, we propose to contribute to this evolution by introducing some formalism from *sampling theory* into numerical simulation of light.

Fast Multipole Methods and Clustering

Clustering approaches have appeared as another promising avenue to simplify the geometry of space. Clustering approaches were inspired by the FMM (Fast Multipole Methods) approach [Gre88], initially proposed as an efficient way of solving the N-Body physical problem. FMM uses a hierarchical decomposition of the computational domain as well as higher-order series of expansions to group sets of sufficiently distant particles into "super-particles". A review of FMM methods applied to global illumination is given in [HDSD99].

[CW93] Michael F. Cohen and John R. Wallace. *Radiosity and Realistic Image Synthesis*. Academic Press Professional, Boston, MA, 1993.

[GTGB84] C. Goral, K.E. Torrance, D. Greenberg, and B. Battaile. Modeling the interaction of light between diffuse surfaces. In *SIGGRAPH*. ACM, 1984.

[HSA91] P. Hanrahan, D. Salzman, and L. Aupperle. A rapid hierarchical radiosity algorithm. In *Computer Graphics (Siggraph)*. ACM, 1991.

[GSCH93] S.J. Gortler, P. Schroder, M.F. Cohen, and P. Hanrahan. Wavelet radiosity. In *Computer Graphics (Siggraph)*. ACM, 1993.

[Sch94] P. Schroeder. *Wavelet Methods for Global Illumination*. PhD thesis, Princeton University, 1994.

[Zat93] H.R. Zatz. Galerkin radiosity: a higher order solution method for global illumination. In *Computer Graphics (Siggraph)*. ACM, 1993.

[Arv95] James Arvo. The role of functional analysis in global illumination. In *Rendering Techniques*. Springer-Verlag, 1995.

[Gre88] L. Greengard. *The rapid evaluation of potential fields in particle systems.* MIT Press, Cambridge, Massachusetts, 1988.

[HDSD99] J.M. Hasenfratz, C. Damez, F. Sillion, and G. Drettakis. A practical analysis of clustering strategies for hierarchical radiosity. In *Computer Graphics Forum*. Eurographics, 1999.

Monte-Carlo Integration

Stochastic methods is another possible family of numerical methods for solving Equation 1. As such, Monte-Carlo integration can solve an integral equation by issuing random samples according to some probabilistic distribution function. The differential operators are re-casted in terms of relations between probability distributions. Historically, to solve the rendering Equation, Kajiya proposed in [Kaj86] the *path tracing* approach, an extension of *distribution ray tracing* [CPC84]. Path tracing can be thought of as a Monte-Carlo sampling of a Neumann series expansion of the Rendering Equation. This approach was improved in [LW93] and [VG94]. They propose an improved estimator, that integrates over light paths from both the eye and the light sources. However, stochastic methods suffer from noise, caused by a high variance of estimation. To reduce this variance, different techniques were proposed. For instance, the *Russian roulette* used in particle physics [SG69] was introduced in Computer Graphics in [AK90]. The Monte-Carlo family of methods became popular these last few years, since they are much simpler to implement than finite element methods, and since they (apparently) require less mathematical background to be understood. Jensen coined the term *Photon Mapping* [Jen01] to qualify Monte-Carlo integration applied to the rendering equation (the randomly chosen samples of light intensity are naturally renamed as "photons"). This contributed to make the method popular, now understood as an extension of the classic Ray-Tracing method. Note also that it was relatively easy to extend photon mapping to take into account directional effects (glossy materials, caustics). Recent work extended the Rendering Equation to take into account even more complex models[JMLH01]: the light can now enter the subsurface of the material and can be scattered multiple times before exiting the material. As a consequence, the ρ Bidirectional Reflectance Distribution Function is replaced by a Bidirectional Surface Scattering Reflectance Distribution Function, a function that depends not only on the directions, but also on the entry and exit points of light. This requires integrating not only over all the possible directions, but also over a surfacic neighborhood around the point under consideration.

Pre-computed Radiance Transfer (PRT)

Recently, guided by applications, a new class of problems has started to be studied by the Computer Graphics community. Instead of computing light simulation in a given scene, the goal is now to pre-compute how an object interacts with its lighting environment (i.e., pre-computing the radiance transfer function - PRT - yielded by the object)[SKS02]. This makes it possible to instantaneously take into account changes in the lighting environment. From the application point of view, it makes it possible to have a library of pre-processed objects, ready to be used in real-time in arbitrary shading environments (for instance, they can be integrated in a movie, during the post-production phase). In a certain sense, this is a natural evolution of early shading models. In those early models, a physically deduced analytic expression models the way the object reacts to lighting. Now, with PRT, the analytic expression is replaced by a representation in certain function bases, numerically computed

[Kaj86] J. Kajiya. The rendering equation. In *Computer Graphics (Siggraph)*, 1986.

[CPC84] Robert L. Cook, Thomas Porter, and Loren Carpenter. Distributed ray tracing. In *SIGGRAPH*, 1984.

[LW93] Eric P. Lafortune and Yves D. Willems. Bidirectional path tracing. In *Compugraphics*, 1993.

[VG94] Eric Veach and Leonidas Guibas. Bidirectional estimators for light transport. In *Eurographics Workshop on Rendering*, 1994.

[SG69] Jerome Spanier and Ely Gelbard. Monte carlo principles and neutron transport problems. In *Reading, MA*. Addisson-Wesley, 1969.

[AK90] James Arvo and David Kirk. Particle transport and image synthesis. In *SIGGRAPH*, 1990.

[Jen01] H.W. Jensen. *Realistic Image Synthesis Using Photon Mapping*. Natik, MA: A. K. Peters, 2001.

[JMLH01] H.W. Jensen, S. R. Marschner, M. Levoy, and P. Hanrahan. A practical model for subsurface light transport. In *Computer Graphics (Siggraph)*. ACM, 2001.

[SKS02] P.-P. Sloan, J. Kautz, and J. Snyder. Pre-computed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. *ACM TOG (Siggraph)*, 2002.

from the object's geometry and photometric properties. PRT computes the light transfer function of an object, i.e., the function that maps source illumination into transferred incident illumination at each point of the object. Spherical Harmonics are used to represent this function, and their coefficients are estimated (e.g. using Monte-Carlo integration). To compress the huge amount of data needed to represent the transfer function, classic methods are used (principal components analysis and its variants). Note that not only the transfer function needs special care, but also the representation of the lighting environment. In [NRH03], a non-linear wavelet approximation is used to represent it, which enables capturing high-frequency / low-frequency interactions, such as complex shadows and highlights.

Open problems

One of the difficulties is finding efficient ways of evaluating the visibility term. This is typically a Computational Geometry problem, i.e., a matter of finding the right combinatorial data structure (the *visibility complex*), studying its complexity and deriving algorithms to construct it. To deal with this issue, several teams (including VEGAS, ARTIS and REVES) study the visibility complex.

The other terms of the Rendering Equation cannot be solved analytically in general. As can be seen in this quick state of the art overview, many different numerical resolution methods have been used. The main difficulties of the discipline is that each time a new physical effect should be simulated, the numerical resolution methods need to be adapted. In the worst case, it is even necessary to design a new ad-hoc numerical resolution method. For instance, in Monte-Carlo based solvers, several sampling maps are used, one for each effect (a map is used for the diffuse part of lighting, another map is used for caustics ...). As a consequence, the discipline becomes a collection of (sometimes mutually exclusive) techniques, where each of these technique can only simulate a specific lighting effect.

The other difficulty is to satisfy two somewhat antinomic objectives at the same time. On the one hand, we want to simulate complex physical phenomena (subsurface scattering, polarization, interferences, ...), responsible for subtle lighting effects. On the other hand, we want to visualize the result of the simulation in real-time.

One of ALICE's research directions is to design new representations of lighting coupled with the geometric representation. These representations of lighting need to be general enough so as to be easily extended when multiple physical phenomena should be simulated. Moreover, we want to be able to use these representations of lighting in the frame of real-time visualization. Our research plans are detailed in Section 6.2 below.

[NRH03] R. Ng, R. Ramamoorthi, and P. Hanrahan. All-frequency shadows using non-linear wavelet lighting approximation. *ACM TOG (Siggraph)*, 2003.

5 Applications: visualization

Participants: *L. Alonso, X. Cavin, B. Lévy, C. Mion, J. Muller, R. Toledo*

Figure 4: *Real-time visual simulation of the Renault Velsatis, using our physically-based light simulator and our Digital Geometry Processing solutions (VSP-Technology).*

Visualization is one the main application domains that require an optimized geometric representation of the objects. With the advances realized in both acquisition processes and numerical simulation methods, larger and larger volumes of data need to be visualized. The solutions we develop help extracting a structure from these large volumes of data and organizing them. The so-constructed structure optimizes the visualization process, and helps interacting with numerical simulations applied to the model.

These data structures can be used to visualize both computer models of manufactured objects (see Figure 4) and natural objects (see Figure 5). Although our representations of geometric objects and their properties initially aim at simulating light-matter interactions, the representations that we develop are *generic*. Therefore, they can be applied to other application domains. For instance, the volume visualization algorithms that we develop [LCCC01] can be used to visualize numerical simulations in geosciences [CLB05], and in plasma physics [CLCP04]. Similarly, our parameterization methods [LPRM02] can be applied in industrial visualization [RUCL03] and in biochemistry [RCPM05]. We cooperate with specialists of these different domains, in the frame of past and present cooperations, including the start-ups earth decisions sciences and VSP-Technology, ACI Geogrid (ministry of research grant), ARC Plasma and ARC Docking (INRIA national grant).

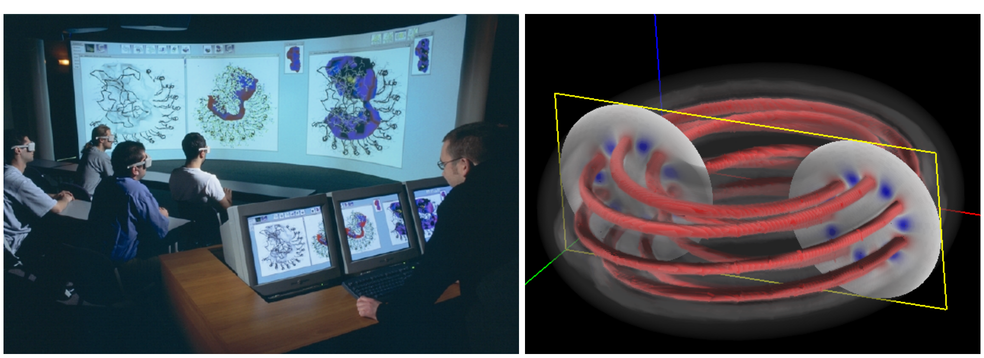


Figure 5: *Left: immersive molecular docking simulation (ARC Docking). Right: visualization of a plasma physics simulation (ARC Plasma).*

6 Research directions

ALICE aims at developing new solutions to represent geometric objects (surfaces, volumes, ...) and physical properties attached to them. Since our main research interest is Computer Graphics, we pay a particular attention to photometric properties.

To study both classes of problems (digital geometry processing and numerical simulation of light), our common approach is composed of the following steps, as classically done in numerical modeling (the originality of our approach is discussed further):

1. formalize the operator as a PDE (Partial Derivatives Equation) or Integral Equation. In the case of light simulation, this equation is a variant of the Rendering Equation (see Section 4.2). As far as geometry processing is concerned, notions from physics can help the intuition. For instance, the expression of the flexion energy of a thin plate can be used to design smoothing operators;

2. discretize the operator in a form compatible with the representation of the object, using numerical schemes such as finite differences, finite elements, Galerkin, ...

3. study the discretized operator, and prove its properties, such as existence and uniqueness of the solution, independence to the discretization, ...

4. design a minimization algorithm, based on Numerical Analysis methods, such as Conjugate Gradient, Gauss-Newton, Lagrange, Galerkin, Monte-Carlo... To keep computation times reasonable, it is possible to use several acceleration techniques, such as pre-conditioners and multi-scale/multi-grid approaches.

We try to design solutions that are *provably correct*, *scalable* and *numerically stable*:

- ◇ by provably correct, we mean that we want to ensure the existence and uniqueness of the solution. In addition, in the case of a meshed model, we try to design methods that are independent of the mesh density. In the case of mesh parameterization, we want to ensure the injectivity of the constructed mapping;
- ◇ by scalable, we mean that our solutions need to be applicable to data sets of industrial size. For instance, for a scanned mesh, 10 millions triangles is now an average;
- ◇ by numerically stable, we mean that our solutions need to be resistant to the degeneracies often encountered in industrial data sets. Note: this characteristic is also named *robustness* in another context. However, "robustness" now refers to a certain class of methods. Most of those methods evaluate the sign of a determinant, using dynamic precision, modular arithmetics, interval arithmetics or filtered predicates ... that avoid the artifacts caused by the limited precision of numbers. In our Numerical Analysis context, the goals are different (find the minimum of an objective function) and the methods are different (Schur complement, pre-conditioners, automatic differentiation, ...), this is the reason why we prefer to use the standard term from Numerical Analysis.

The next sections give our research directions for applying our approach to Digital Geometry Processing (Section 6.1) and to the numerical simulation of light (Section 6.2). We then present in Section 6.3 our longer-term research plans, common to both disciplines.

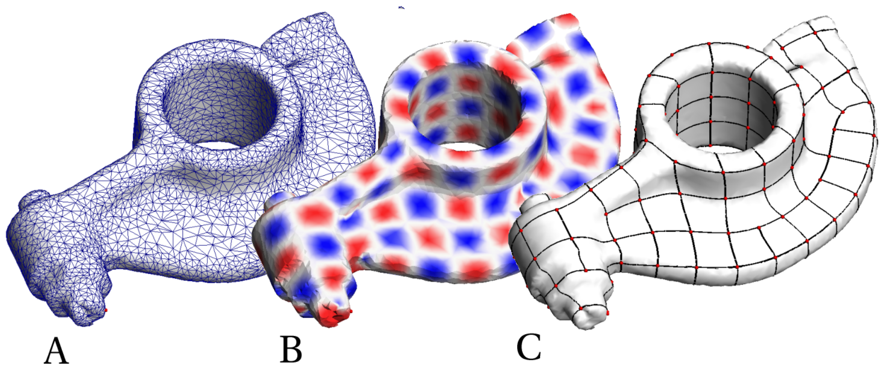


Figure 6: *A: a scanned mesh (mechanical part); B: our function defined over the mesh; C: converting the mesh into a T-Spline surface*

6.1 Digital Geometry Processing

Participants: *L. Alonso, B. Lévy, W.C. Li, N. Ray, B. Vallet, B. Wang*

As explained in Section 4.1, 3D shapes are represented by two categories of models: meshed objects and the ”curves and surfaces” family of representations. One of our objectives is to design a new solution to automatically convert a meshed model into a “curves and surfaces” representation. Note that a solution to this difficult problem would immediately have applications to many other Computer Graphics problems, including texture mapping, visualization of large models, remeshing and grid generation.

Automatic conversion from mesh to ”curves and surfaces” representations is a tedious geometry processing problem due to their fundamental difference. While a mesh is a discrete representation of the geometry by enumeration and sampling, ”curves and surfaces” describes the geometry in a continuous way by equations: $x = x(u,v), y = y(u,v), z = z(u,v)$. While one can easily discretize continuous data by using adequate sampling, the reverse process is much more difficult. This reverse problem is identified of one of the difficult open problems in Digital Geometry Processing. In a nutshell, this process requires three steps:

1. **segmentation:** decompose the object into a set of topological discs (called charts), see e.g. [CSAD04];
2. **parameterization:** parameterize each chart in a domain in 2D, which means that every single 3D point on the mesh can be represented by a point in the 2D domain and the correspondence is one-to-one;
3. **Spline fitting:** one may notice that after the parameterization, we obtain the same form of representation of the geometry as in the ”curves and surfaces” representation, i.e., in the form of three functions $x(u,v), y(u,v)$ and $z(u,v)$. In a Spline, those three functions are themselves parameterized by their degree, control points and knot. Therefore, by adequately choosing the values of degree, control points and knot of the spline, the original mesh can be now approximated by the spline with a certain approximation error between the target spline surface and the original mesh.

[CSAD04] D. Cohen-Steiner, P. Alliez, and M. Desbrun. Variational shape approximation. In *Computer Graphics (Siggraph)*. ACM, 2004.

ISA is an early contributors to the Digital Geometry Processing discipline, and more particularly to the problem of mesh parameterization [LM98], [Lev01], [LPRM02], [Lev03]. We currently develop a new functional optimization method, that constructs a segmentation (step 1) and a parameterization (step 2) at the same time (see Figure 6). The method constructs two functions defined over the surface, characterized by their gradients aligned with the directions of principal curvatures. The so-defined function is a parameterization almost everywhere, except in the vicinity of zero-points (or Morse points), as predicted by general topology. To minimize deformations further, unlike other *globally smooth* parameterization methods[GY02], we add the possibility of relaxing the injectivity condition. This makes it possible to construct a quasi-isometric parameterization, at the expense of creating more singular points than required by Euler-Poincarre theorem. This quasi-isometric parameterization is well-suited to fitting methods applied to Splines and subdivision methods (its isometric property minimizes unwanted oscillations).

Besides Spline fitting, the functional representation of surfaces constructed by our approach can have other applications. Texture mapping is the most obvious application: the 2D-parameter space is painted with an image, transformed onto the object by the parameterization. In addition, fine-scale geometric details can be stored at a low memory cost in parameter-space, which makes it possible to simplify the geometric model (this technique is known as 'bump mapping'). This representation can be displayed in real time by computer graphics hardware. This wide possible spectrum of possible applications (spline fitting, remeshing, texture mapping, visualization ...) is the reason why parameterization has became a highly competitive domain in the Digital Geometry Processing community. Our goal is now to try to forecast what will be the evolution of this topic, based on the following remark: until recently, Computer Graphics only required a *surfacic* representation of the objects. In the context of light simulation, early lighting models used to only consider light-matter interaction at the surface of the objects. As mentioned in the previous section, more sophisticated models for the light-matter interaction have recently been introduced. In those models, light enters inside the material and is scattered multiple times. Simulating these phenomena requires a *volumic* representation of the objects (see e.g. [CTW+04]). For this reason, we think that in the next few years, the emerging research topic related with those volumic representations will have the same development as surface parameterization. To construct these 3D representations, we plan to develop a simple 3D generalization of our functional representation.

Originality of our approach to Digital Geometry Processing: Early approaches to Digital Geometry Processing were driven by a Signal Processing approach. In other words, the solution of the problem is obtained after applying a *filtering scheme* multiple times. This is for instance the case of the mesh smoothing operator defined by Taubin in his pioneering work[Tau95]. Recent approaches still inherit from this background. Even if the general trend moves to Numerical Analysis, much work in Digital Geometry Processing still study the coefficients of the gradient of the objective function *one by one*. This intrinsicly refers to *descent* methods (e.g. Gauss-Seidel), which are not the most efficient, and do not converge in general when applied to meshes larger than a certain size (30000 facets). To extend the feasibility of the methods, solvers based on a multi-resolution setting are often used. These multi-resolution methods first solve the equation on a coarse approximation of the object. This solves the equation for the low frequencies of the signals. The geometric details are iteratively added, and the solution is iteratively updated, by interleaved descent iterations.

[GY02] X. Gu and S.-T. Yau. Computing conformal structures of surfaces. *Communications in information and systems*, 2:121–146, 2002.

[CTW+04] Yanyun Chen, Xin Tong, Jiaping Wang, Stephen Lin, Baining Guo, and Heung-Yeung Shum. Shell texture functions. *ACM TOG(SIGGRAPH)*, 2004.

[Tau95] G. Taubin. A signal processing approach to fair surface design. In *SIGGRAPH Conference Proceedings*, pages 351–358. ACM, 1995.

This progressively injects the higher frequencies in the solution. The main drawback is the difficulty to tune the parameters introduced by these methods (number of resolution levels, number of iterations used at each level and convergence criterion). The other drawback is the difficulty of implementation of the multi-resolution data structure and the associated multi-resolution analysis that separate the spacial frequencies of the geometric objects.

In our approach, Digital Geometry Processing is systematically restated as a (possibly non-linear and/or constrained) functional optimization problem. As a consequence, studying the properties of the minimum is easier: the minimizer of a multivariate function can be more easily characterized than the limit of multiple applications of a smoothing operator. This simple remark makes it possible to derive properties (existence and uniqueness of the minimum, injectivity of a parameterization, and independence to the mesh). For instance, our Least Squares Conformal Maps method [LPRM02] minimizes a provably positive definite quadratic form. An equivalent formula was simultaneously discovered by Desbrun[DMSB99], based on a much more intuitive geometric setting. However, Desbrun's formula appeared in a form that was more difficult to study and to minimize (they did not prove the existence and unicity of the minimizer). However, since both methods do not always construct an injective mapping, we designed later in cooperation with A. Sheffer (University of British Columbia) a new algorithm[SLMB04] that does not suffer from this limitation.

Besides helping to characterize the solution, restating the geometric problem as a numerical optimization problem has another benefit. It makes it possible to design efficient numerical optimization methods, instead of the iterative relaxations used in classic methods. For instance, in addition to correctness, the method we developped in [SLMB04] constructs a distortion minimizing parameterization for a large mesh (up to 4 million triangles). The method is based on the constrained minimization of a quadratic functional by the Newton-Lagrange algorithm. Some algebraic manipulations of the Hessian matrix reduce the size of the linear systems and dramatically improves performances. In addition, some algebraic manipulations ensure the numerical stability of the methods, by minimizing the number of divisions appearing in the formula.

6.2 Numerical Simulation of light

Participants: *L. Alonso, X. Cavin, G. Lecot, B. Lévy, B. Vallet*

As explained in Section 4.2, the numerical simulation of light means solving for light intensity in the Rendering Equation. The solution of this equation is characterized by smooth variations almost everywhere, except near shadow boundaries and highlights. The two main difficulties are the problem of *representation* and the problem of *sampling*:

- ◇ **representation:** define a function space in which the solution of the Rendering Equation can be approximated using a minimum quantity of information (i.e., a minimum number of coefficients);
- ◇ **sampling:** given a fixed number of basis functions, choose the elements of a function basis so as to obtain the best approximation of the solution.

To solve the rendering equation, two categories of methods have been studied:

- ◇ **Finite element methods:** the Rendering Equation is projected onto a linear function basis (Galerkin method), adaptively refined. Finite Element methods efficiently compute the overall smooth variations of the solution, only a small number of coefficients is required to represent the solution in smooth zones. In other words, Finite Element Methods

[DMSB99] M. Desbrun, M. Meyer, P. Schröder, and A.H. Barr. Implicit fairing of irregular meshes using diffusion and curvature flow. In *SIGGRAPH Conference Proceedings*, pages 317–324. ACM, 1999.

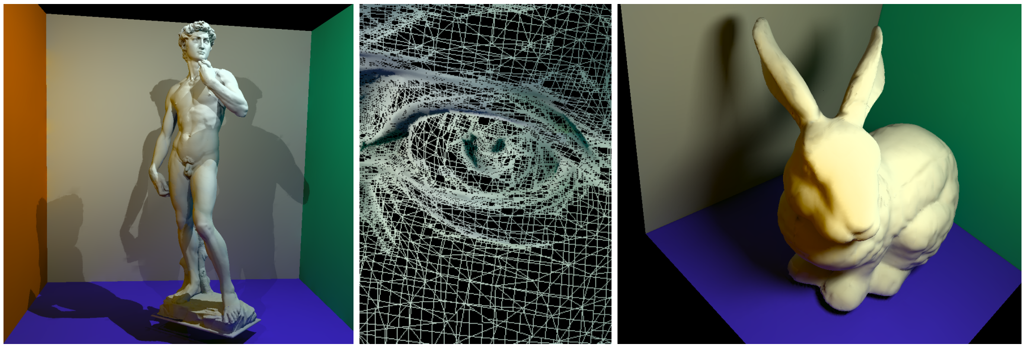


Figure 7: *Left: Discrete Master Element global illumination applied to "David" (500K Δ). Center: DME decouples the representation of the lighting from the geometry. Sharp shadows are accurately captured by adaptive wavelets, represented in parameter-space. The image shows the geometric triangulated mesh with the wavelets FEM control mesh superimposed; Right: DME Stanford Bunny with area lights (70K Δ).*

perform well from a representation point of view. However, a large number of subdivisions is required to accurately capture shadow boundaries or highlights (see e.g. Figure 8). In other words, finding the right sampling is a difficult part of the problem. Note that generating finite elements aligned with shadow boundaries (discontinuity meshing), as we have done in [HA04] (e.g. finding a better sampling) dramatically reduces the required number of subdivisions;

- ◇ **Stochastic methods:** the solution is represented by a set of random samples, generated according to a density of probability that characterizes the terms of the Rendering Equation. The situation is somewhat opposed to finite element methods: the unstructured set of samples used by these methods manage to capture some small shading details of high intensity, such as caustics, but are slow to converge in large uniform zones. The presence of *noise* is also an important problem with these methods. Note also that sharp variations will be only captured if they correspond to high intensities of light, since the obtained sampling density is proportional to the value of the lighting signal. As a consequence, shadow boundaries are generally poorly sampled by those approaches. However, this class of numerical methods have much popularity in the Computer Graphics community due to their ease of implementation, their similarities with the well known ray-tracing algorithm, and the "photon" analogy coined by Jensen[Jen01].

Our short-term research plans concern several evolutions of Finite Element Methods and Monte-Carlo sampling. In a longer term, we plan to develop new solution mechanisms, based on an adaptive unstructured sampling with high-order elements.

Research topics in Finite Element Methods

The numerical simulation of light based on Finite-Elements was one of the research axes of the ISA project. The main contribution of ISA in this discipline is the *Virtual Mesh* approach [ACP$^+$01], which decouples the representation of light from the representation of geometry. In the initial Virtual Mesh method, we considered that the geometry of the scene was represented by quadric surfaces. We are currently developing a discrete Master Element method, to make the method applicable to large tessellated meshes. Our method uses a parameterization to decouple the representation of the energy from the geometry of the scene. Both can be *independently* and *continuously* considered at different scales, which optimizes large-scale transfers and enables capturing sub-facet lighting details. We plan to experiment our approach combined with various finite-element global illumination

[Jen01] H.W. Jensen. *Realistic Image Synthesis Using Photon Mapping*. Natik, MA: A. K. Peters, 2001.

methods, including hierarchical Galerkin radiosity simulations, and pre-computed radiosity transfers.

Research topics in Stochastic Methods

As explained in Section 4.2, recent advances in stochastic light simulation concern two principal aspects: one direction of research deals with taking into account complex phenomena, such as light scattering in translucent materials [JMLH01]. Recent advances[CTW+04] consider translucent materials with heterogeneities distributed over a set of layers. In other words, surfaces are replaced with "thick" surfaces, with photometric properties that vary across the thickness (the previous section outlines our research plans to construct those volumic representations). The other direction develops Precomputed Radiance Transfer methods (see Section 4.2) that can be used in a real-time rendering context.

The interest and the applications of Precomputed Radiance Transfer are obvious (natural generalization of lighting models, library of synthetic objects that can be directly integrated in movies at the post-production phase). However, if the lighting conditions and materials reach a certain degree of complexity, we think that the objectives of Precomputed Radiance Transfer are too ambitious and cannot be fulfilled with a reasonable amount of storage. For this reason, we plan to solve for the radiance in the Rendering Equation under *fixed lighting conditions*. This has two consequences:

- ⋄ We no longer need the distant lighting assumption. Light sources are free to be placed at arbitrary locations in the scene, including near the objects;
- ⋄ Instead of needing to store a *functional* at each point of the scene, we just need to store a *function*. In a Spherical Harmonic basis, we need to store a vector instead of a matrix. This means that with a fixed amount of storage, higher order spherical harmonics can be used, and more subtle lighting effects can be captured.

To develop this radiance simulation method, our short term plans are as follows: our results in mesh parameterization make it possible to attach spherical harmonic coefficients to each point of a scene represented by a mesh. These coefficients can be computed by a two-phases algorithm: Monte-Carlo sampling, and gathering in parameter space. From an intuitive point of view, instead of computing an image that covers the screen, we compute an image that covers the scene (and each pixel of this image is in fact a function that depends on the viewing angle).

Longer term research topics

As previously mentioned, in most works in Computer Graphics, Monte-Carlo sampling is currently applied to the Rendering Equation with the "photons" image in head. We think that this artificially restricts the class of representations and methods that can be used. In the "photon mapping" approach, the solution of the equation is represented by a set of discrete samples, and the sampling density is proportional to the value of the function. Sophisticated filtering schemes can reconstruct a smooth function from this discrete sampling during the gathering phase. We think that the speed of convergence of the algorithm may be dramatically improved by using higher order samples, representing not only the value of the radiance at a given point, but also its derivatives. The interaction of these samples with the objects will also require higher order lighting models, involving not only the normal vector, but also the tensor of curvature at the point under consideration. Our goal is to replace as much stochastic computations as possible with symbolic computations. The samples are

---

[JMLH01] H.W. Jensen, S. R. Marschner, M. Levoy, and P. Hanrahan. A practical model for subsurface light transport. In *Computer Graphics (Siggraph)*. ACM, 2001.

[CTW+04] Yanyun Chen, Xin Tong, Jiaping Wang, Stephen Lin, Baining Guo, and Heung-Yeung Shum. Shell texture functions. *ACM TOG(SIGGRAPH)*, 2004.

then no longer a single value, but a set of coefficients in some function bases, representing a larger amount of information. Monte-Carlo sampling is then applied in an abstract, higher dimensional space, yielded by those function bases. If we stick to the "photons" image, our approach may be thought of as a multi-scale method, that considers not only photons, but also physical quantities defined at a coarser scale. For instance a "flux" emerges from the contribution of a large number of photons (like in thermodynamics: temperature, entropy, enthalpy emerge from the contribution of a large number of particles).

More generally, as explained in the next section, we aim at studying both the solution mechanism and the problem of sampling. In a dynamic environment, where objects and lights can change, it would be possible to define a dynamic representation of the lighting, continuously enforcing the Rendering Equation and ensuring that an optimum sampling of its solution is constructed. Our plans to study those general problems, related to both Digital Geometry Processing and Numerical Light Simulation axes, are detailed in the next section.

Originality of our approach to Numerical Light Simulation: Richard Feynman (Nobel Prize in physics) mentions in his lectures that physical models are a "smoothed" version of reality. The global behavior and interaction of multiple particles is captured by physical entities of a larger scale. According to Feynman, the striking similarities between equations governing various physical phenomena (e.g. Navier-Stokes in fluid dynamics and Maxwell in electromagnetism) is an illusion that comes from the way the phenomena are modeled and represented by "smoothed" larger-scale values (i.e., *fluxes* in the case of fluids and electromagnetism). Note that those larger-scale values do not necessarily directly correspond to a physical intuition, they can reside in a more abstract "computational" space. For instance, representing lighting by the coefficients of a finite element is a first step in this direction. More generally, our approach consists in trying to get rid of the limits imposed by the classic view of the existing solution mechanisms, that come from a physical intuition. Instead of trying to mimic the physical process, we try to restate the problem as an abstract numerical computation problem, on which more sophisticated methods can be applied (a plane flies like a bird, but it does not flap its wings). From this point of view, our approach to the problem of Numerical Light Simulation is the same as what we do in Digital Geometry Processing. We try to consider the problem from a computational point of view, and focus on the link between the numerical simulation process and the properties of the solution of the Rendering Equation.

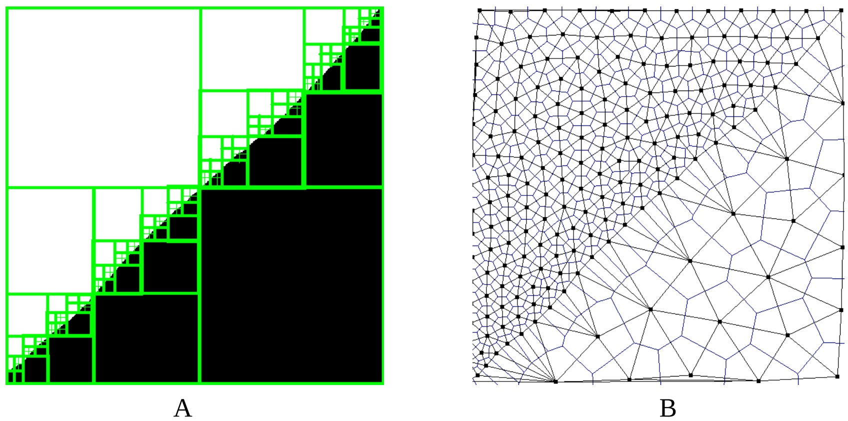


Figure 8: *A: representing a function using a wavelet basis; B: the same function sampled by a weighted centroidal Voronoi diagram, using the same number of function bases. A more accurate approximation is obtained.*

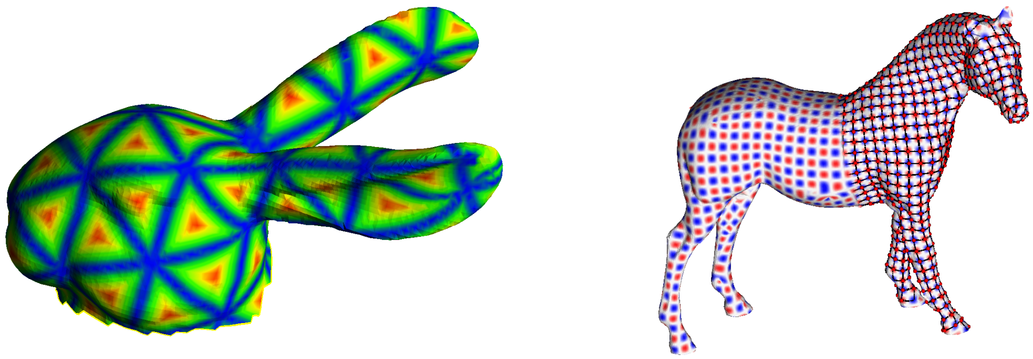
6.3 Numerical Approximation

Participants:*L. Alonso, B. Lévy, W.C. Li, N. Ray, B. Vallet, B. Wang*

This section describes our longer term research directions, for problems that are common to the "Digital Geometry Processing" and the "Simulation of light" research axes.

The general class of problems we are interested in may be formalized as follows: given a function f (note: f can be either explicitly given, or defined to be the solution of a Partial Differential or Integral Equation), given a linear space of functions B_k, find the vector x such that $\tilde{f} = \sum x_k B_k$ is the best approximation of f (e.g. relative to the L^2 norm $\int(f - \tilde{f})^2$). The case where f is explicitly given corresponds for instance to Spline fitting problems. The case where f is the solution of a differential equation corresponds for instance to Galerkin-based global light simulation. In classic methods, the function basis B_k is fixed. For instance, the functions B_k correspond to finite elements defined over the cells of a mesh. Adaptive refinement relaxes this condition, and make it possible to make the solution more precise where high spatial frequencies are encountered. As such, global illumination based on wavelets can adaptively refine the function basis where sharp variations of lighting occur. Figure 8-A shows how a wavelet basis can adapt discontinuities of a signal. As can be seen on the figure, the anisotropy of the arbitrarily chosen axes have a great influence on the result, and a large number of refinements is required when the orientation of the elements does not match the orientation of the discontinuities.

For this reason, we propose to study a more general formulation of the problem: the function is still approximated by a basis B_k, but all the functions of B_k depend on a vector of *unknown* parameters p. Now we want to find the vector x and the vector p such that $\tilde{f} = \sum x_k B_k(p)$ is the best approximation of f. For instance, suppose that the function $\tilde{f}$ is a bivariate piecewise linear function, defined on the triangles of a Delaunay triangulation. There is one basis function B_k per vertex k of the triangulation, and the vector of parameters p corresponds to the coordinates of all the vertices of the triangulation. Optimizing for both x and p means finding the best approximation *and the best sampling* at the same time. Figure 8-B shows how a discontinuity can be approximated by this type of function basis. The case of fixed parameters p is well known in approximation theory (and corresponds to least-squares fitting). The case where the parameters p are not fixed is still open in general. For instance, in PDE solving, this means finding the optimum mesh and the coefficients of the finite elements on this mesh at the same time. In spline-fitting problems, this means optimizing both the topology of the control mesh and the location of the vertices



A B

Figure 9: *Continuous remeshing: the remeshing problem is restated as a continuous functional optimization problem, either with three functions to generate triangles (A), or two functions to generate quads (B).*

of the control mesh at the same time. Since the way the basis functions B_k depend on the parameters p is non-trivial, classic numerical optimization methods cannot be applied. In our example, the supports of the basis functions B_k depend on the combinatorics of the Delaunay triangulation of the vertices p. This yields discontinuities of the derivatives of the B_k functions relative to p that prevent classic optimization methods from being applied.

From Digital Signal Processing theory, we know some solutions to construct an optimum sampling of a given signal. Lloyd's relaxation method[Llo82] iteratively updates the locations of a set of sites (in our context, they correspond to the p parameters) until their configuration converges to a Centroidal Voronoi Diagram, provably optimum from a sampling point of view. Convergence is ensured under certain convexity conditions. Cohen-Steiner and Alliez successfully applied this approach to various geometric approximation problems (see e.g. [CSAD04]). However, from a computational point of view, Lloyd's relaxation is very similar to the Gauss-Seidel iterative method for solving a linear system. In the context of numerical optimization, we know that some algorithms (e.g. Conjugate Gradient) are much more efficient than Gauss-Seidel's method. One of our goals is to find counterparts of those efficient numerical methods in the frame of optimum sampling problems. To attack this general problem, we plan to try the following approaches:

1. **Discrete optimization:** The most ambitious way of dealing with this problem that we consider can be outlined as follows: Conjugate Gradient relies on the analytic notion of derivative and on conjugacy, a certain notion of orthogonality. It may be possible in our context to define similar notions that are algebraically well behaved, i.e., that satisfy the set of axioms required by the Conjugate Gradient method. This would replace Lloyd's relaxation with a more efficient "discrete Conjugate Gradient". If we do not succeed with this approach, we plan to explore two other solutions, outlined below;

2. **Prolongated discrete optimization:** if the previous approach is not feasible, another possibility is to consider a continuous function that interpolates our unknown discrete function, express an objective function defined in a way that its minimizer satisfies the discrete optimality condition, and extract the discrete solution from the continuous minimizer. Figure 9 shows our first experiments to apply this idea to remeshing problems. The mesh emerges as the zero-set of a continuous function that satisfies some optimality conditions. The right part of Figure 9-B shows the extracted mesh;

[Llo82] S. Lloyd. Least square quantization in PCM. *IEEE Trans. Inform. Theory*, 28:129–137, 1982.

[CSAD04] D. Cohen-Steiner, P. Alliez, and M. Desbrun. Variational shape approximation. In *Computer Graphics (Siggraph)*. ACM, 2004.

3. **Mixed numerical/symbolic approaches:** to solve these difficult optimization problems, another possibility is to develop tools that manipulate both numeric and symbolic representations of the function at the same tile. The symbolic representation can be dynamically transformed, the expression of the derivatives can be formally computed, and numerical algorithms adapted to the properties of the function can be dynamically selected during the optimization process. This approach could be explored in cooperation with the SPACES and GALAAD Inria project teams (SPACES studies ways of solving systems of algebraic equations, and GALAAD studies algebraic geometry). Parallel solvers is also a possible research avenue to keep computation times reasonable. The team has contributed several methods for the parallelization of dynamic irregular problems in general [CAP98, Cav99, CA00], and more particularly for radiosity simulations based on hierarchical finite elements .

7 Software

7.1 Software developed by the team

Graphite is a research platform for computer graphics, 3D modeling and numerical geometry. It comprises all the main research results of the last three years from our "Numerical Geometry" group. Data structures for cellular complexes, parameterization, multi-resolution analysis and numerical optimization are the main features of the software. Graphite is publicly available (GPL license) since October 2003, and is now used by researchers from Geometrica (INRIA Sophia Antipolis), Artis (INRIA Grenoble), LSIIT (Strasbourg), Technion (Israel), Stanford University (United States), Harvard University (United States), University of British Columbia (Canada) .... Graphite is one of the common software platforms that will be used in the frame of the European Network of Excellence AIM at Shape. Within Graphite, **OpenNL** is a numerical library well adapted to the sparse numerical problems encountered in geometry processing. OpenNL includes several iterative solvers (conjugate gradient, bicgstab, gmres) and several preconditioners (Jacobi, SSOR). OpenNL can also wrap other solvers (SuperLU, MUMPS).

LightSim is an experimental light simulator, based on software components from Candela (the finite element solver developed by ISA) and Graphite. LightSim is currently under development. It will contain the code corresponding to our experiments described in section 6.2.

THC/Gigaviz is a visualization software used in our cooperations with researchers from other domains (plasma physics and biochemistry). A component under development will allow visualizing large data sets on PC clusters. It uses components from Graphite and LightSim.

7.2 Other software

CGAL is a collaborative effort of several sites in Europe and Israel, originally funded by the Information Technologies program Esprit of the European Union. CGAL provides robust implementations of computational geometry algorithms. ALICE uses the 2D and 3D Delaunay triangulations of CGAL. We currently develop a CGAL parameterization package based on OpenNL is in cooperation with GEOMETRICA, in the frame of the AIM at Shape European Network of Excellence.

SUPERLU and MUMPS are solvers for large sparse systems of linear equations. SuperLU is developed by the University of Berkeley, and MUMPS is developed by the GRAAL

INRIA project. We cooperate with J.-Y. L'Excellent (GRAAL) in the frame of our ARC GEOREP.

8 Expected results and criteria of success

In this project proposal, our goal is to develop new solutions and demonstrate their feasibility in two highly competitive domains of Computer Graphics: Digital Geometry Processing and Light Simulation. Digital Geometry Processing recently emerged, and is now a quickly evolving field with a lot of scientific activity (the number of articles per year related to this field doubled or tripled these last few years), and Light Simulation remains one of the fundamental aspects of Computer Graphics. Our first expected outcome is the publication of our results in the principal conferences and journals of the domain (SIGGRAPH, ACM Transactions on Graphics, IEEE Visualization, IEEE Transactions on Visualization and Computer Graphics). Since we experiments applications of our methods with specialists of other disciplines, we also target journals like IEEE Computer Graphics and Applications, Numerical Geology and Journal of Computational Chemistry.

From the industrial point of view, in a longer term perspective, the techniques we plan to develop also answer to unsatisfied needs. We previously transfered our constrained griding techniques to the Earth Decision Sciences start-up, which now hires more than 100 employees in 5 different countries. As far as digital geometry processing is concerned, in the industry, most of the processing is done manually. Our work in light simulation, parameterization and texture mapping has already been successfully transferred to the industry, by the VSP-Tech company who commercializes Candela and who created the new X-Mesh product. We think that developing a new technology that analyzes and converts geometry, that does realistic light simulation in real time, may continue in the same direction and is very likely to have an impact in the industry in the long term.

9 Project-team positioning

9.1 Positioning in the Computer Graphics community

Our two areas of research (light simulation and digital geometry processing) are very competitive. Since it is one of the fundamental aspects of Computer Graphics, light simulation has always been an active area of research. Digital Geometry Processing recently emerged and has shown an exponential activity these last few years. The institutes interested in both disciplines include Caltech (Schroeder and Desbrun), Technion (Gotsmann), Harvard (Gortler), University of British Columbia (Sheffer), Stanford, Microsoft Research (Hoppe), Microsoft Research Asia.

The members of the project are early contributors to the Digital Geometry Processing discipline [LM98], [Lev01], [LPRM02], [Lev03]. In light simulation, they have made some significant contributions [ACP$^+$01]. In both disciplines, our specificity consists in using a well-suited formalism (numerical analysis) on top of which we design general solutions. In this highly moving and competitive context, we alternatively cooperate and compete with the actors of these two disciplines.

9.2 Positioning with respect to other INRIA projects

ARTIS: We have the same overall objectives as ARTIS: find the best representation adapted to a specific problem, with the difference that ARTIS attacks a large class of problems, with a wide spectrum of methods. This is also the case of other Inria Computer Graphics project teams (REVES, EVASION). We prefer to focus on a more limited set of problems, requiring longer-term research, but likely to have a long-term impact.

GEOMETRICA: In his recent evolution, PRISME became GEOMETRICA and now besides computational geometry, GEOMETRICA also includes Digital Geometry Processing (and more specifically *approximation*) in his research plans. There is some overlapping between the different problems we work on. For some of those problems, we cooperate (e.g. we co-develop a mesh parameterization package for CGAL, we worked together on anisotropic polygonal remeshing [ACSD+03]), for some other problems we develop alternative methods in parallel. The main difference is the original background we use (GEOMETRICA uses Computational Geometry and we use Numerical Analysis). For instance, as far as *approximation* is concerned, GEOMETRICA uses the geometric properties of centroidal voronoïdiagrams and we use the analytical properties of an objective function.

VEGAS: We share some objectives with VEGAS. As explained in Section 3.1, the difference is that VEGAS restricts his studies to problems for which a closed form can be found. In our case, we consider general geometric problems, for which a closed form does not always exist. As a consequence, we use a different set of theoretical tools (numerical analysis), and our scientific communities are different: Computational Geometry for VEGAS and Computer Graphics,Digital Geometry Processing for ALICE.

10 Cooperations

10.1 Cooperations with other INRIA project-teams

In the frame of our ARC DOCKING, we cooperate with the INRIA projects GEOMETRICA and EVASION.

In the frame of our ARC GEOREP, we cooperate with the INRIA projects GRAAL (J.-Y. L'Excellent), MOVI (E. Boyer), ARTIS (X. Decoret, G. Debunne and E. Eisemann). The research theme concerns the full geometric data processing pipeline, from acquisition to numerical simulation.

We keep strong relations with the other spin-offs of ISA (VEGAS and MAGRITTE). In particular, our light simulation research axis may benefit from the studies on the visibility complex carried on by VEGAS.

We participate to the ARC SHOW (ARTIS), on the representation and manipulation of large geometric models.

We cooperate on a regular basis with members of the GEOMETRICA project (P. Alliez) on Digital Geometry Processing [ACSD+03], and with members of the ARTIS project (N. Holschutz) on numerical light simulation [HA04].

10.2 Cooperations with other French research groups

The University of Nancy - Edam (molecular dynamics) is a member of our ARC Docking.

The University of Strasbourg - LSIIT (J.-M. Dischler) is a member of our ARC GEOREP. We cooperate on a regular basis, on texture, parameterization and light simulation [DMLC02].

10.3 Cooperations with foreign research groups

We are a member of the `AIM@Shape` European Network of Excellence.

The University of British Columbia - Imager Lab (A. Sheffer) is a member of our ARC GEOREP. We cooperate on a regular basis, on mesh parameterization and digital geometry processing [SLMB04].

We cooperate with the Beckmann Institute (USA), on the VMD molecular simulation and visualization software (16000 users in the world). We developed the *Intersurf* plugin in the frame of our ARC Docking.

10.4 Industrial cooperations

We cooperate with the VSP-Tech company, a spin-off of the ISA project. We recently patented and transferred a new parameterization technique.

Laurent Castanié (Cifre Earth Decisions Sciences) is co-advised by B. Lévy.

We proposed our *Geometry Intelligence* proposal to Microsoft Research's "tools for advancing science" call for proposal.

Appendix

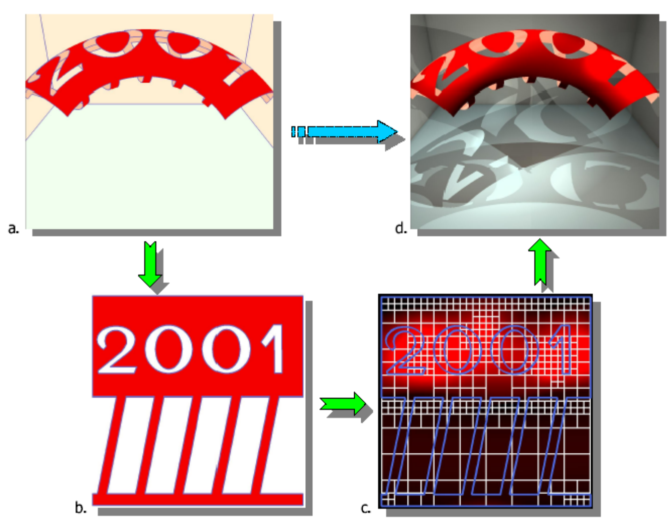
This appendix summarizes the most significant papers by members of the project.

The Virtual Mesh : A Geometric Abstraction for Efficiently Computing Radiosity

ACM Transactions on Graphics - 2001

L. Alonso, F. Cuny, S. Petitjean, J.-C. Paul, S. Lazard and E. Wies

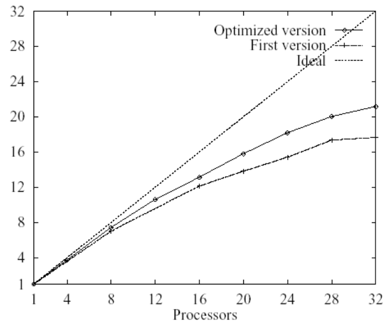
In this paper, we introduce a general-purpose method for computing radiosity on scenes made of parametric surfaces with arbitrary trimming curves. By contrast with past approaches that require a tessellation of the input surfaces (be it made up of triangles or patches with simple trimming curves) or some form of geometric approximation, our method takes fully advantage of the rich and compact mathematical representation of objects. At its core lies the virtual mesh, an abstraction of the input geometry (Figure a) that allows complex shapes to be illuminated as if they were simple primitives. The virtual mesh is a collection of normalized square domains (Figure b) to which the input surfaces are mapped while preserving their energy properties. Radiosity values are then computed on these supports before being lifted back to the original surfaces (Figure c). To demonstrate the power of our method, we describe a high-order wavelet radiosity implementation that uses the virtual mesh. Examples of objects and environments, designed for interactive applications or virtual reality, are presented. They prove that, by exactly integrating curved surfaces in the resolution process, the virtual mesh allows complex scenes to be rendered more quickly, more accurately and much more naturally than with previously known methods (Figure d).

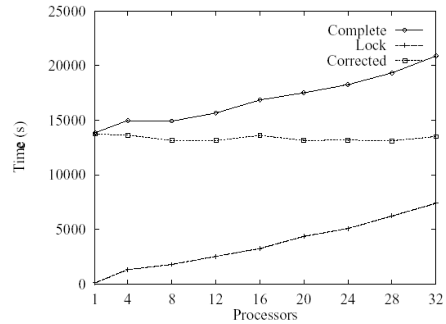


Partitioning and Scheduling Large Radiosity Computations in Parallel

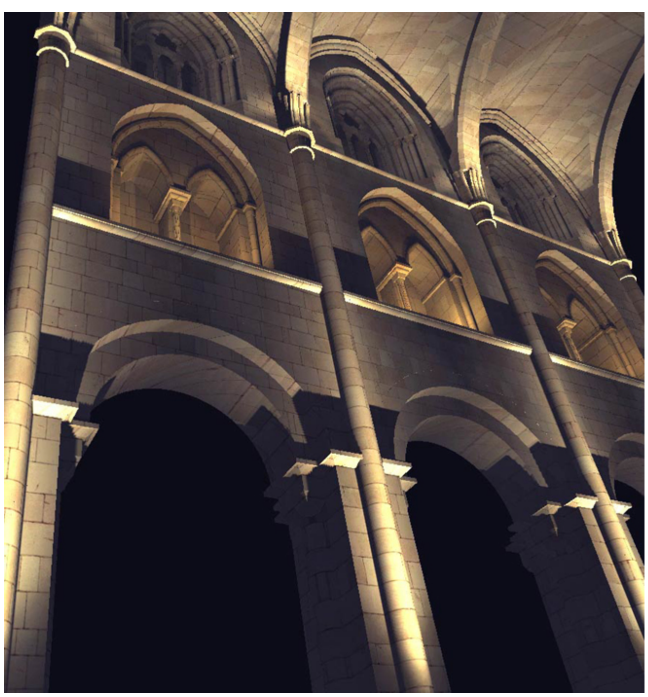
Journal on Parallel and Distributed Computer Practices - 2000

X. Cavin, J.-C. Paul and L. Alonso





We show, in this paper, how it is feasible to efficiently perform large radiosity computations on a conventional (distributed) shared memory multiprocessor machine. Hierarchical radiosity algorithms, although computationally expensive, are an efficient view-independent way to compute the global illumination which gives the visual ambiance to a scene. Their effective parallelization is made challenging, however, by their non-uniform, dynamically changing characteristics, and their need for long-range communication. To address this need, we have developed appropriate partitioning and scheduling techniques, that deliver an optimal load balancing, while still exhibiting excellent data locality. We provide the detailed implementation of these techniques applied to our Virtual Mesh algorithm, and present results of experiments showing very good acceleration and scalability performances. The left curves show the obtained acceleration as a function of the number of processors. The right curve shows the cumulated time of all processors, and the time spent in critical sections. The accurate radiosity solutions required to render high quality images of an extremely large model are computed in a reasonable time (see image below). The rendering capabilities of modern graphics hardware are then used to visualize this virtual pre-lit environment in real-time.



Constrained Texture Mapping
SIGGRAPH 2001
B. Lévy

Recently, time and effort have been devoted to automatic texture mapping. It is possible to study the parameterization function and to describe the texture mapping process in terms of a functional optimization problem. Several methods of this type have been proposed to minimize deformations. However, these existing methods suffer from several limitations. For instance, it is difficult to put details of the texture in correspondence with features of the model, since most of the existing methods can only constrain iso-parametric curves. We introduce in this paper a new optimization-based method for parameterizing polygonal meshes with minimum deformations, while enabling the user to interactively define and edit a set of constraints. Each user-defined constraint consists of a relation linking a 3D point picked on the surface and a 2D point of the texture. Our method constructs a parameterization X by minimizing the following objective function:

$$C(X) \quad = \quad \sum_{i=1}^{m}(M_i - X(U_i)) + \varepsilon \int_{\Omega} \left(\frac{\partial^2 X}{\partial u^2}\right)^2 + \left(\frac{\partial^2 X}{\partial v^2}\right)^2 dudv$$

where the (M_i, U_i)'s are the constrained points and where ε makes it possible to choose a trade-off between the smoothness of the solution and the data fitting.

The non-deformation criterion introduced here can act as an extrapolator, thus making it unnecessary to constrain the border of the surface, in contrast with classic methods. To minimize the criterion, a conjugate gradient algorithm is combined with a compressed representation of sparse matrices, making it possible to achieve a fast convergence.

Least Squares Conformal Maps

SIGGRAPH 2002

B. Lévy, S. Petitjean, N. Ray and J. Maillot

A Texture Atlas is an efficient color representation for 3D Paint Systems. The model to be textured is decomposed into charts homeomorphic to discs, each chart is parameterized, and the unfolded charts are packed in texture space. Existing texture atlas methods for triangulated surfaces suffer from several limitations, requiring them to generate a large number of small charts with simple borders. The discontinuities between the charts cause artifacts, and make it difficult to paint large areas with regular patterns.

In this paper, our main contribution is a new quasi-conformal parameterization method, based on a least-squares approximation of the Cauchy-Riemann equations. We minimize the conformal energy, defined on each triangle by:

$$C(T) = \int_T \left| \frac{\partial \mathcal{U}}{\partial x} + i \frac{\partial \mathcal{U}}{\partial y} \right|^2 dA = \left| \frac{\partial \mathcal{U}}{\partial x} + i \frac{\partial \mathcal{U}}{\partial y} \right|^2 A_T,$$

where A_T is the area of the triangle and the notation $|z|$ stands for the modulus of the complex number z. We now suppose that the parameterization $\mathcal{U}$ is piecewise linear. The gradient is constant over each triangle, and its two components can be gathered in a complex number:

$$\frac{\partial u}{\partial x} + i \frac{\partial u}{\partial y} = \frac{i}{d_T} \begin{pmatrix} W_1 & W_2 & W_3 \end{pmatrix} \begin{pmatrix} u_1 & u_2 & u_3 \end{pmatrix}^\top,$$

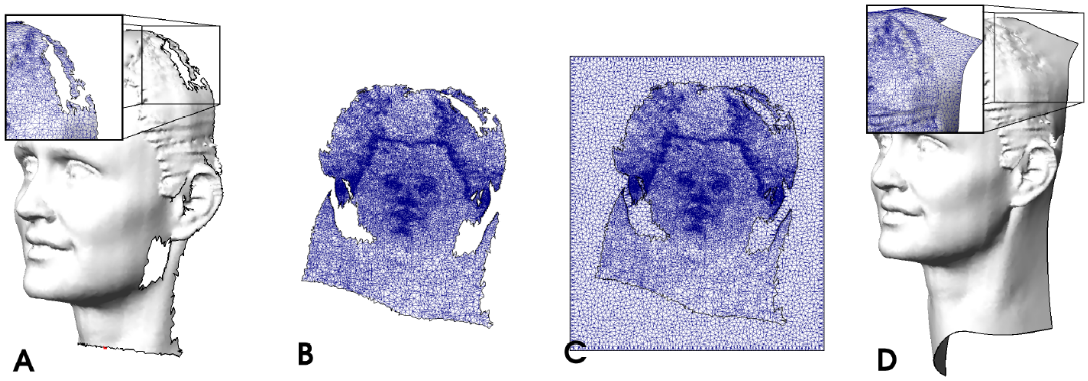
where

$$\left\{ \begin{array}{lcl} W_1 & = & (x_3 - x_2) + i(y_3 - y_2), \\ W_2 & = & (x_1 - x_3) + i(y_1 - y_3), \\ W_3 & = & (x_2 - x_1) + i(y_2 - y_1). \end{array} \right.$$

The objective function defined by summing the conformal energy over all the triangles of a surface minimizes angle deformations, and we prove the following properties: the minimum is unique, independent of a similarity in texture space and independent of the resolution of the mesh. The function is numerically well behaved and can therefore be very efficiently minimized. Our approach is robust, and can parameterize large charts with complex borders.

We also introduce segmentation methods to decompose the model into charts with natural shapes, and a new packing algorithm to gather them in texture space. We demonstrate our approach applied to paint both scanned and modeled data sets.

Dual Domain Extrapolation
SIGGRAPH 2003
B. Lévy



Shape optimization and surface fairing for polygon meshes have been active research areas for the last few years. Existing approaches either require the border of the surface to be fixed, or are only applicable to closed surfaces. In this paper, we propose a new approach, that computes natural boundaries. This makes it possible not only to smooth an existing geometry, but also to extrapolate its shape beyond the existing border. Our approach is based on a global parameterization of the surface and on a minimization of the squared curvatures, discretized on the edges of the surface. Using a global parameterization makes it possible to completely decouple the outer fairness (surface smoothness) from the inner fairness (mesh quality). In addition, the parameter space provides the user with a new means of controlling the shape of the surface. When used as a geometry filter, our approach computes a smoothed mesh that is discrete conformal to the original one. This allows smoothing textured meshes without introducing distortions. Our approach is outlined in the figure: Due to shadows, scanned meshes often have complex holes and irregular borders (Figure A). A global parameterization of the surface is computed (Figure B). Filling the holes and extrapolating the borders become 2D problems in parameter space (Figure C). The locations of the new vertices in 3D space are computed by approximating a minimal energy surface (Figure D).

The surface constructed by our approach is an approximation of a minimal energy surface (MES), defined by the following energy functional:

$$E_{MES} \quad = \quad \int_{\Omega} \kappa_{min}^2 + \kappa_{max}^2 \, dudv$$

We propose a discrete version of the energy functional, defined as follows:

$$E_{MES} \simeq F(x) = \frac{1}{6} \sum_{e \in \mathscr{E}} \mathscr{A}(T) + \mathscr{A}(T') \left\| \frac{J_T.\mathbf{w}_2(e)}{\|J_T.\mathbf{w}_2(e)\|} - \frac{J_{T'}.\mathbf{w}_2(e)}{\|J_{T'}.\mathbf{w}_2(e)\|} \right\|^2$$

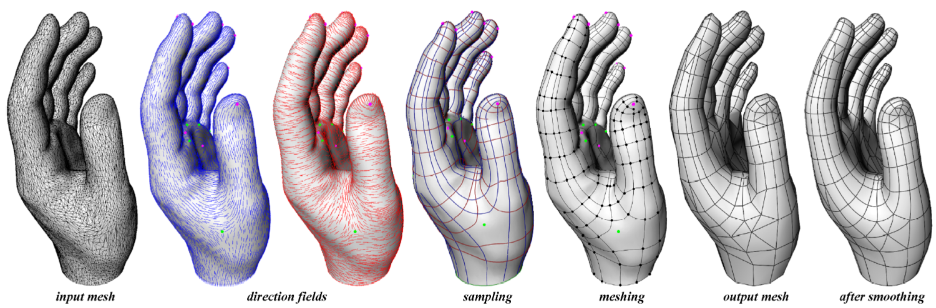
$$e = (i,j) \quad ; \quad \mathbf{w}_2(e) = \begin{bmatrix} v_i - v_j \\ u_j - u_i \end{bmatrix}$$

where $J(\delta)$ denotes the Jacobian matrix of $\mathbf{x}(.,.)$ (i.e. the matrix of the differential $d\mathbf{x}$) at the point $\mathbf{u} + \delta.\mathbf{w}$, given by:

$$J \quad = \quad \begin{pmatrix} \partial x/\partial u & \partial y/\partial u & \partial z/\partial u \\ \partial x/\partial v & \partial y/\partial v & \partial z/\partial v \end{pmatrix}^t \qquad (2)$$

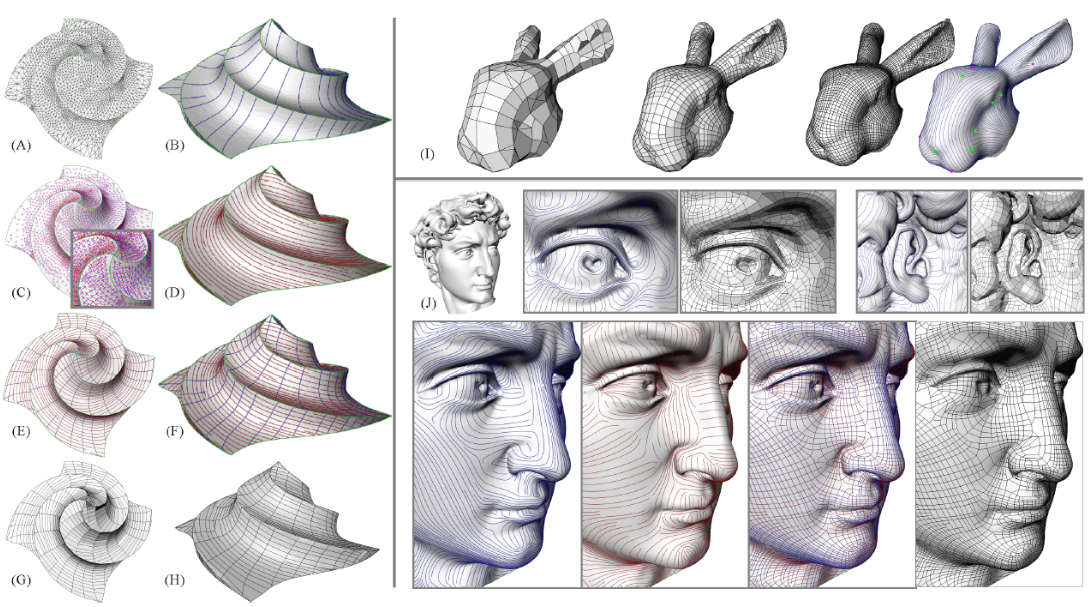
To minimize this non-linear fairing functional, we experiment different solvers, including Newton's method, BFGS, and SQP (sequential quadratic programming).

Anisotropic Polygonal Remeshing
SIGGRAPH 2003
P. Alliez, O. Devillers, B. Lévy and M. Desbrun



In this paper, we propose a novel polygonal remeshing technique that exploits a key aspect of surfaces : the intrinsic anisotropy of natural or man-made geometry. In particular, we use curvature directions to drive the remeshing process, mimicking the lines that artists themselves would use when creating 3D models from scratch. After extracting and smoothing the curvature tensor field of an input geometry patch, lines of minimum and maximum curvatures are used to determine appropriate edges for the remeshed version in anisotropic regions, while spherical regions are simply point-sampled since there is no natural direction of symmetry locally. As a result our technique generates polygon meshes mainly composed of quads in anisotropic regions, and of triangles in spherical regions. Our approach provides the flexibility to produce meshes ranging from isotropic to anisotropic, from coarse to dense, and from uniform to curvature adapted.

The figure below shows our method applied to various data sets. A-H: the octa-flower geometry illustrates the behavior of our remeshing technique for piecewise smooth surfaces. Principal direction fields are estimated and piecewise smoothed (C) I: The bunny's head is remeshed with different mesh densities. J: Finally, Michelangelo's David is remeshed; close-ups on the eye and the ear show the complexity of the model, and how the lines of curvatures match the local structures. Below is another closeup, on the whole face this time, with lines of curvatures and nal polygonal mesh.



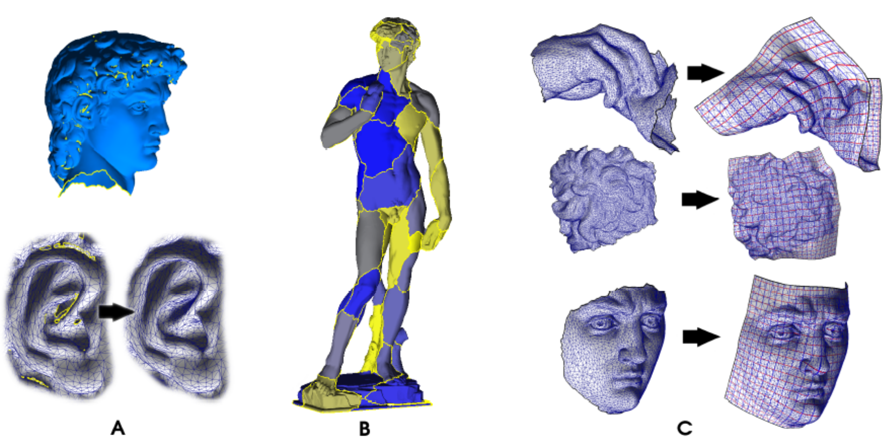
Master Element Radiosity for Large Tesselated Models
Submitted
G. Lecot, B. Lévy, L. Alonso and J.-C. Paul

Complete numerical simulation of global illumination is a computationally intensive process. The complexity of the problem has two main aspects:

- ◇ **geometric complexity:** Highly Tesselated models dramatically increase computation times;
- ◇ **lighting complexity:** Sharp shadows and small-scale lighting variations are difficult to represent.

For these two reasons, finite element methods perform poorly when applied to highly Tesselated models, due to the tight coupling they introduce between the geometry and the lighting.

In this paper, we introduce the Parametric Master Element (PME), a new structure that makes finite-element methods applicable to large tessellated meshes, by using a parameterization to decouple the representation of the energy from the geometry of the scene. Both can be *independently* and *continuously* considered at different scales, which optimizes large-scale transfers and enables capturing sub-facet lighting details. PME may be used by various finite-element global illumination methods. The view-independent solution, represented in parameter-space, can be directly exploited by graphics hardware for interactive walkthroughs. To construct our representation, we first repair the model (Figure A), then we decompose the model into charts (Figure B), then we parameterize and extrapolate each chart (Figure C). The radiosity equation is then solved by a method similar to our Virtual Mesh method.



11 Project-team bibliography

References

[ACP$^+$01] L. Alonso, F. Cuny, S. Petitjean, J.-C. Paul, S. Lazard, E. Wies, "The Virtual Mesh : A Geometric Abstraction for Efficiently Computing Radiosity", *ACM Transactions on Graphics 20*, 3, Jul 2001, p. 169–201.

[ACSD$^+$03] P. Alliez, D. Cohen-Steiner, O. Devillers, B. Levy, M. Desbrun, "Anisotropic Polygonal Remeshing", *ACM Transactions on Graphics (SIGGRAPH) 22*, 3, Jul 2003, p. 485–493, SIGGRAPH 2003 Session : Surfaces.

[ACUP00] L. Alonso, X. Cavin, J.-C. Ulysse, J.-C. Paul, "Fast and Accurate Wavelet Radiosity Computations Using High-End Platforms", *in : Third Eurographics Workshop on Parallel Graphics & Visualisation, Girona, Spain*, U. de Girona (editor), p. 25–38, Sep 2000.

[CA00] X. Cavin, L. Alonso, "Parallel Management of Large Dynamic Shared Memory Space : a Hierarchical FEM Application", *in : Seventh International Workshop On Solving Irregularly Structured Problems In Parallel - Irregular'2000, Cancun, Mexico*, J. R. et al. (editor), *Lecture Notes in Computer Science*, *1800*, S. Sahni, T. Davis, S. Rajasekeran, S. Ranka, Springer, p. 428–434, May 2000.

[CAH00] F. Cuny, L. Alonso, N. Holzschuch, "A novel approach makes higher order wavelets really efficient for radiosity", *Computer Graphics Forum 19*, 3, Sep 2000, p. C99–108, (Eurographics 2000 proceedings).

[CAP98] X. Cavin, L. Alonso, J.-C. Paul, "Parallel Wavelet Radiosity", *in : Proceedings of the Second Eurographics Workshop on Parallel Graphics and Visualisation , Rennes, France*, T. P. Kadi Bouatouch, Alan Chalmers (editor), p. 61–75, sep 1998.

[CAP99] X. Cavin, L. Alonso, J.-C. Paul, "Overlapping Multi-Processing and Graphics Hardware Acceleration : Performance Evaluation", *in : Parallel Visualization and Graphics Symposium 1999, San Francisco, CA*, A. SIGGRAPH (editor), IEEE, p. 79–88, Oct 1999.

[Cav99] X. Cavin, "Load Balancing Analysis of a Parallel Hierarchical Algorithm on the Origin2000", *in : Fifth European SGI/Cray MPP Workshop , Bologna, Italy*, CINECA, Oct 1999.

[CLB05] L. Castanie, B. Levy, F. Bosquet, "VolumeExplorer: Roaming Large Volumes to Couple Visualization and Data Processing for Oil and Gas Exploration", *in : Visualization conf. proc., to appear*, IEEE, 2005.

[CLCP04] G. Caumon, B. Levy, L. Castanie, J.-C. Paul, "Advanced visualization for various unstructured grids using ah hoc topological structures.", *Computers and Geosciences*, Sep 2004.

[CMF05] X. Cavin, C. Mion, A. Filbois, "COTS Cluster-based Sort-last Rendering: Performance Evaluation and Pipelined Implementation", *in : Visualization conf. proc., to appear*, IEEE, 2005.

[CPA00] X. Cavin, J.-C. Paul, L. Alonso, "Partitioning and Scheduling Large Radiosity Computations in Parallel", *Journal on Parallel and Distributed Computer Practices - Special Issue on Parallel and Distributed Computer Graphics 3*, 3, Sep 2000.

[DMLC02] J.-M. DICHLER, K. MARITAUD, B. LEVY, D. CHAZANFARPOUR, "Texture Particles", *in : Eurographics 2002 - EG 2002, Saarbrcken, Germany*, Sep 2002.

[HA00] N. HOLZSCHUCH, L. ALONSO, "Using graphics hardware to speed-up your visibility queries", *Journal of Graphics Tools 5*, 2, 2000, p. 33–47.

[HA04] N. HOLZSCHUCH, L. ALONSO, "Combining Higher-Order Wavelets and Discontinuity Meshing : a Compact Representation for Radiosity", *in : Eurographics Symposium on Rendering 2004 - EGSR 2004, Norrkping, Sweden*, p. 13, Jun 2004.

[HCA00] N. HOLZSCHUCH, F. CUNY, L. ALONSO, "Wavelet Radiosity on Arbitrary Planar Surfaces", *in : 11th Eurographics Workshop on Rendering, Brno, rpublique tchque*, B. Peroche, H. Rushmeier (editors), *Rendering Techniques*, Springer-Verlag, Wien, Jun 2000.

[LCCC01] B. LÉVY, G. CAUMON, S. CONREAUX, X. CAVIN, "Circular Incident Edge Lists : A Data Structure for Rendering Complex Unstructured Grids", *in : IEEE Visualization 2001, San-Diego, USA*, IEEE, Oct 2001.

[Lev01] B. LEVY, "Constrained Texture Mapping", *in : SIGGRAPH*, ACM, Addison Wesley, Aug 2001.

[Lev03] B. LEVY, "Dual Domain Extrapolation", *ACM Transactions on Graphics (SIGGRAPH) 22*, 3, Jul 2003, p. 364–369, SIGGRAPH 2003 Session : Parameterization.

[LLP05] W. C. LI, B. LEVY, J.-C. PAUL, "Mesh Editing with an Embedded Network of Curves", *in : Shape Modeling International conference proceedings*, 2005.

[LM98] B. LÉVY, J.-L. MALLET, "Non-Distorted Texture Mapping For Sheared Triangulated Meshes", *in : SIGGRAPH*, ACM, 1998.

[LP02] B. LEVY, S. PETITJEAN, "Least Squares Conformal Maps", *in : Fifth International Conference on Curves and Surfaces 2002 , Saint-Malo, France*, Jun 2002.

[LPRM02] B. LEVY, S. PETITJEAN, N. RAY, J. MAILLOT, "Least Squares Conformal Maps for Automatic Texture Atlas Generation", *in : SIGGRAPH*, ACM (editor), Jul 2002.

[Ray03] N. RAY, *Génération automatique d'atlas de texture*, Thèse d'université, Nancy 2, Apr 2003.

[RCNL05] N. RAY, X. CAVIN, T. NEIGER, B. LEVY, "Vector Texture Maps", *in : Pacific Graphic*, 2005.

[RCPM05] N. RAY, X. CAVIN, J.-C. PAUL, B. MAIGRET, "Intersurf: dynamic interface between proteins", *Journal of Molecular Graphics and Modelling 23*, 4, 2005.

[RL03] N. RAY, B. LEVY, "Hierarchical Least Squares Conformal Maps", *in : 11th Pacific Conference on Computer Graphics and Applications 2003 - PG'03, Canmore, Canada*, p. 263–270, Octobre 2003.

[RUCL03] N. RAY, J.-C. ULYSSE, X. CAVIN, B. LEVY, "Generation of Radiosity Texture Atlas for Realistic Real-Time Rendering", *in : Eurographics conf. proc.*, 2003.

[SLMB04] A. Sheffer, B. Levy, M. Mogilnitsky, A. Bogomyakov, "ABF++: Fast and Robust Angle Based Flattening", *ACM Transactions on Graphics*, Apr 2004.