VNU Journal of Science, Natural Sciences and Technology 24 (2008) 179-185

Moving parabolic approximation model of point clouds and its application

Zhouwang Yang[1], Tae-wan Kim[2*]

[1]*Department of Naval Architecture and Ocean Engineering Seoul National University, Seoul 151-744, Korea*

[2]*Department of Naval Architecture and Ocean Engineering and Research Institute of Marine Systems Engineering, Seoul National University, Seoul 151-744, Korea*

Received 31 October 2007

Abstract. We propose the moving parabolic approximation (MPA) model to reconstruct an improved point-based surface implied by an unorganized point cloud, while also estimating the differential properties of the underlying surface. We present examples which show that our reconstructions of the surface, and estimates of normal and curvature information, are accurate for precise point clouds and robust in the presence of noise. As an application, our proposed model is used to generate triangular meshes approximating point clouds.

1. Introduction

Acquiring large amounts of point data from real objects has become more convenient because of modern sensing technologies and digital scanning devices. However, the data acquired is usually distorted by noise, arising out of physical measurement processes, and by the limitations of the acquisition technologies. Even so, it is possible to obtain the smooth underlying shapes which are implied by an unstructured point cloud. Consequently, techniques of reconstructing models from noisy data sets are receiving increasing attention. Point-based surfaces [1-3] have recently become an appealing shape representation in computer graphics and can be used for geometric modeling [4]. The point-based representation of a surface should be as compact as possible, meaning that it is neither noisy nor redundant. It is therefore important to develop algorithms which generate compact point sets from nonuniform and noisy input, so as effectively to reconstruct the underlying surfaces. It should also be possible to recover the intrinsic geometric properties of the underlying surfaces as precisely as possible from point clouds.

Differential quantities such as normals, principal curvatures, and principal directions of curvature can be used for a variety of tasks in computer graphics, computer vision, computer-aided design, geometric modeling, computational geometry, and industrial and biomedical engineering. A number of methods for curvature estimation have been published by

* Corresponding author. Email: taewan@snu.ac.kr

various communities, but mostly for manifold representations of the surface such as polyhedral meshes, or oriented data sets such as points paired with normals. We would like to recover the differential properties of an underlying surface directly from an unstructured point cloud, even though it may be nonuniform and noisy. Our approach, motivated by some recent work of Levin [2], is based on local maps of differential geometry [5] and practical algorithms in optimization theory [6]. The main contribution of this work is a scheme to generate a point-based reconstruction of an unorganized point cloud and simultaneously to estimate the differential properties of the underlying surface. As an application, we will used the proposed technique to reconstruct triangular meshes approximating given point clouds.

2. Moving parabolic approximation

Recently, there has been increasing interest expressed in surface modeling using unorganized data points. A powerful approach is the use of the moving least-squares (MLS) technique [2] for modeling point-based surfaces [1]. One of the main strengths of MLS projection is its ability to handle noisy data. We extend the MLS technique to a moving parabolic approximation (MPA), which is a model of a second-order projection. The MPA model is naturally framed as an optimization problem based on the following proposition:

Proposition 1: At every point p on a surface S , there exists an osculating paraboloid $\mathcal{S}_p^*$ such that the normal curvature of $\mathcal{S}_p^*$ is identical to that of S at $\mathbf{p}$ for any tangent vector.

2.1. MPA model

Suppose that a given set of data points $\{p_j\}_{j=1}^{n}$ is noisy sampling of an underlying surface S. Generally, $\mathbf{p}j$ will not lie on the underlying shape S due to noise. We first define a neighborhood of the given point cloud in the form:

$$\mathcal{B}(r) = \bigcup_{j=1}^{n} \left\{ \mathbf{x} \in \mathbb{R}^3 \mid \left\| \mathbf{x} - \mathbf{p}_j \right\| \le r \right\} \quad (1)$$

With an assumption

$$r \ge \max_{j1} \min_{j2 \ne j1} \left\| \mathbf{p}_{j_1} - \mathbf{p}_{j_2} \right\| \quad (2)$$

we ensure that the neighborhood $\mathcal{B}(r)$ contains the underlying surface as well as the approximation that we are going to construct. A number of points in this neighborhood are chosen for reference, called reference points, which will be projected on to the underlying surface using MPA models.

Let $\mathbf{x} \in \mathcal{B}(r)$ be a reference point in the close neighborhood of the given data points. The foot-point of x on the underlying surface S is denoted as

$$\mathbf{o}_\mathbf{x} = \mathbf{x} + \zeta \mathbf{n}, \quad (3)$$

where n is the unit normal to $\mathcal{S}$, and ζ is the signed distance from $\mathbf{x}$ to $\mathbf{o}_\mathbf{x}$ along $\mathbf{n}$. We aim to compute the foot-point $\mathbf{o}_\mathbf{x}$ and the differential quantities at the foot-point. Let $\{\mathbf{t}_1(\mathbf{n}), \mathbf{t}_2(\mathbf{n})\}$ be the perpendicular unit basis vectors of the tangent plane, so that $\{\mathbf{o}_\mathbf{x}; \mathbf{t}_1, \mathbf{t}_2, \mathbf{n}\}$ forms a local orthogonal coordinate system. Writing $\mathbf{q}_j = \mathbf{p}_j - \mathbf{x}$, we formulate the moving parabolic approximation model as a constrained optimization:

$$\min f(\mathbf{n}, \zeta, a, b, c) = \sum_{j=1}^{n} [\mathbf{q}_j^T \mathbf{n} - \zeta - \frac{1}{2}\left(a\left(\mathbf{q}_j^T \mathbf{t}_1\right)^2 + 2b\left(\mathbf{q}_j^T \mathbf{t}_1\right)\left(\mathbf{q}_j^T \mathbf{t}_2\right) + c\left(\mathbf{q}_j^T \mathbf{t}_2\right)^2\right)]^2 e^{-\frac{\left\|q_j - \zeta n\right\|^2}{\rho^2}} \quad (4)$$

where $(\mathbf{n}, \zeta, a, b, c)$ are decision variables and ρ is a scale parameter.

Once the optimum solution $(\mathbf{n}^*, \zeta^*, a^*, b^*, c^*)$ of the MPA model of Equation (4) has been obtained, we can recover the differential quantities of the underlying surface $\mathcal{S}$ at the foot-point $\mathbf{o}_x = \mathbf{x} + \zeta^*\mathbf{n}^*$, including the principal curvatures and the principal directions of curvature. An osculating paraboloid of the underlying surface at o_x can then be represented by the parametric expression

$$S^*(u,v) = \left(u, v, \frac{1}{2}\left(a^* u^2 + 2b^* uv + c^* v^2\right)\right)^T, \quad (5)$$

in the local coordinate system $\{\mathbf{o}_x; \mathbf{t}_1^*, \mathbf{t}_2^*, \mathbf{n}^*\}$. The first fundamental form of $\mathcal{S}^*(u,v)$ is given by

$$I^* = E\mathrm{d}u^2 + 2F\mathrm{d}u\mathrm{d}v + G\mathrm{d}v^2, \quad (6)$$

where $E=1$, $F=0$ and $G=1$ at the foot-point $\mathbf{o}_x$. The second form of $\mathcal{S}^*(u,v)$ is given by

$$II^* = L\mathrm{d}u^2 + 2M\mathrm{d}u\mathrm{d}v + N\mathrm{d}v^2, \quad (7)$$

where $L = a^*$, $M = b^*$ and $N = c^*$. The mean curvature H^* and the Gaussian curvature K^* can now be calculated as follows:

$$H^* = \frac{LG - 2MF + NE}{2\left(EG - F^2\right)} = \frac{a^* + c^*}{2}, \quad (8)$$

$$K^* = \frac{LN - M^2}{EG - F^2} = a^* c^* - b^{*2}.$$

From this calculation and **Proposition 1**, we obtain the minimum and maximum principal curvatures of the underlying surface $\mathcal{S}$ at $\mathbf{o}_x$:

$$\begin{cases} \kappa_{min}^* = H^* - \sqrt{H^{*2} - K^*}, \\ \kappa_{max}^* = H^* + \sqrt{H^{*2} - K^*}. \end{cases} \quad (9)$$

and the corresponding principal directions of curvature in the tangent plane:

$$\mathbf{e}_{min}^* = \mathbf{t}_1^*, e_{max}^* = \mathbf{t}_2^*, if \kappa_{min}^* = a^* \le c^* = \kappa_{max}^*;$$

$$\mathbf{e}_{min}^* = \frac{b^* t_1^* + \left(\kappa_{min}^* - a^*\right) t_2^*}{\sqrt{\left(\kappa_{min}^* - a^*\right)^2 + b^{*2}}}, \quad (10)$$

$$\mathbf{e}_{max}^* = \frac{\left(\kappa_{max}^* - c^*\right) t_1^* + b^* t_2^*}{\sqrt{\left(\kappa_{max}^* - c^*\right)^2 + b^{*2}}}, \text{otherwise}.$$

The principal directions $\mathbf{e}_{min}^*$ and $\mathbf{e}_{max}^*$ are always orthogonal to each other except at the umbilical points. At an umbilic, $\kappa_{min}^* = \kappa_{max}^*$ holds, and the surface is locally part of sphere with a radius of $1/H^*$. In the special case where the identical principal curvatures vanish, the surface becomes locally flat.

2.2. Implementation and examples

The MPA model of Equation (4) is a constrained optimization problem. We solve this constrained optimization by a practical algorithm based on Lagrange-Newton method [6]. We implement our MPA approach and perform it on a number of point clouds.

The moving parabolic approximation model was tested on several different shapes of surface. Each shape is a graph of a bivariate function $z(x,y)$ defined over $[-1,1] \times [-1,1]$ and evaluated using a 41×41 grid.

$$(x_l, y_k) = (-1 + l/20, -1 + k/20),\ k = 0,\ldots,40,$$

to determine a set of clean points that lie on the graph:

$$\mathcal{P}_{clean} = \left\{\left(x_l, y_k, z\left(x_l, y_k\right)\right)^T \mid l, k = 0, \ldots, 40\right\}$$

In order to verify the stability of the algorithm, we generated a point cloud $\mathcal{P}_{noise}$ by adding Gaussian noise with a magnitude of 1% of the overall cloud dimension to clean data. The four test surfaces were a sphere $(x,y,z)^T = \left(x, y, \sqrt{4 - x^2 - y^2}\right)^T$, a cylinder $(x,y,z)^T = \left(x, y, \sqrt{2 - x^2}\right)^T$, a paraboloid $(x,y,z)^T = \left(x, y, x^2 + y^2\right)^T$, and a hyperboloid $(x,y,z)^T = \left(x, y, x^2 - y^2\right)^T$. The estimated curvature information obtained from MPA model was compared with the exact curvatures in each case. We measured the difference in terms of root-mean-square (RMS) error, which we define as

$$Err = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(val_i^{est} - val_i^{ex}\right)^2}, \qquad (11)$$

where val_i^{est} represents one of the estimated values κ_{min}^{est}, $k_{\max}^{est}$, H^{est} or K^{est}, and val_i^{ex} represents one of the exact values κ_{min}^{ex}, κ_{max}^{ex}, H^{ex} or K^{ex}. Table I summarizes the RMS errors that occurred in the estimation of principal, mean and Gaussian curvatures. From which, we observe that our MPA algorithm can obtain robust and accurate estimates in the presence of noise as well as for clean data.

We also applied the MPA algorithm to the scanning data set of a mouse which contains 36036 points, and presented the point-based reconstruction and the estimates of curvature in Figure 1. The results show the confidence of our MPA method for reverse engineering applications.

Table 1. RMS errors in curvature estimation for the test surfaces

Example	Err (κ_{min})	Err (κ_{max})	Err (H)	Err (K)
Sphere (clean data)	0.0028	0.0014	0.0019	0.0019
(with 1% noise)	0.0412	0.0264	0.0233	0.0238
Cylinder (clean data)	0.0038	3.5e-07	0.0019	2.5e07
(with 1% noise)	0.0747	0.0281	0.0446	0.0215
Paraboloid (clean data)	0.0144	0.0188	0.0158	0.0287
(with 1% noise)	0.0957	0.1075	0.0885	0.1828
Hyperboloid (clean data)	0.0117	0.0017	0.0028	0.0138
(with 1% noise)	0.1278	0.1297	0.0684	0.1505

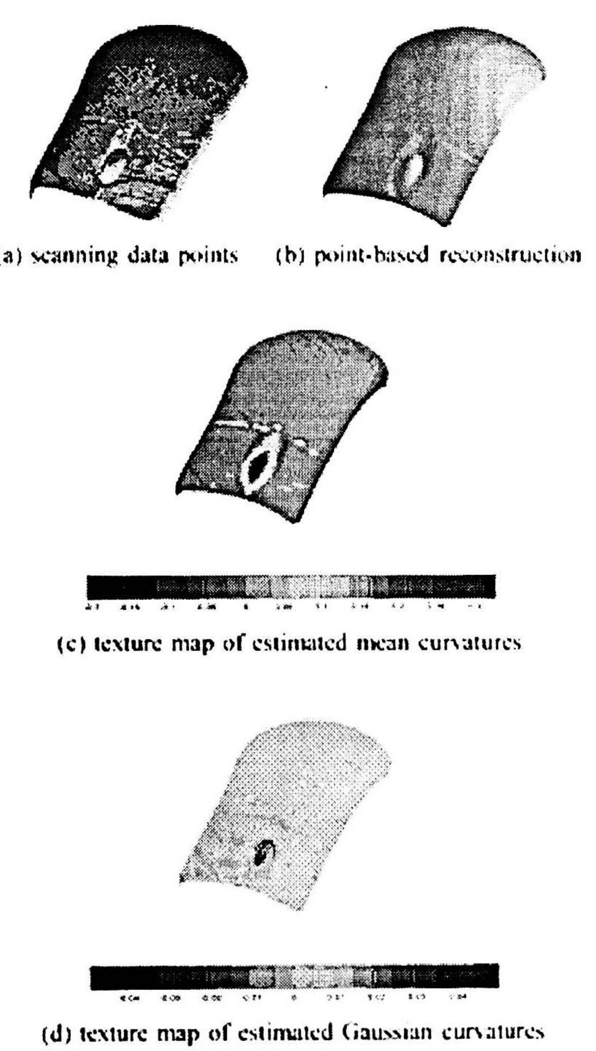


Fig. 1. Applying the MPA algorithm to the Mouse model.

3. Mesh reconstruction

As an application, our MPA model is used to generate a triangular mesh that approximates the underlying surface of given point cloud. Our method of mesh reconstruction from point clouds by moving parabolic approximation can be outlined in the following scheme.

1. A rough initial mesh $\mathcal{M}^{(0)} = (V^{(0)}, E^{(0)})$ is constructed from given point cloud $\mathcal{P} = \{\mathbf{p}_j\}_{j=1}^{n} \subset \mathbb{R}^3$. Let $V^{New} := V^{(0)}$ be the initial set of new inserting vertices.

2. Repeatedly apply the steps of curvature-based refinement (a-b-c) until the approximation error is within a predefined tolerance or the maximal number of times is reached:
   a. For each $\mathbf{v}^N \in V^{New}$, we project it on to the underlying surface of the point cloud P using the MPA algorithm, and get the estimate of mean curvature vector $\mathbf{K}_{\mathcal{P}}(\mathbf{v})$ at the projection $\mathbf{v}$ = MPA($\mathbf{v}^N$). After projection, the set of potential vertices is denoted by

$$V^{Potential} = \left\{ \mathbf{v} = MPA\left(\mathbf{v}^N\right) \middle| \forall \mathbf{v}^N \in V^{New} \text{and} \left\| K_{\mathcal{P}}(\mathbf{v}) \right\| \rangle \sigma \right\}$$

   b. Calculate the mean curvature normal $\mathbf{K}_{\mathcal{M}}(\mathbf{v})$ via the differential geometry operator [7], and define

$$V^{Active} = \{ \mathbf{v} \in V^{Potential} \left\| \mathbf{K}_{\mathcal{M}}(\mathbf{v}) - \mathbf{K}_{\mathcal{P}}(\mathbf{v}) \right\| \rangle \varepsilon \left\| \mathbf{K}_{\mathcal{P}}(\mathbf{v}) \right\| \}$$

   as the collection of active vertices.
   c. Insert a new vertex at the midpoint of every edge adjacent to any $\mathbf{v} \notin V^{Active}$, and then renew $V^{New} = \left\{ \mathbf{v}^N = \frac{\mathbf{v} + \mathbf{v}_i}{2} | \forall\ \mathbf{v} \in \mathrm{V}^{\text{Active}} \right\}$ and $\mathbf{v}\mathbf{v}_i \in E$ }. The approximating mesh is updated by adding the topological connections for those new inserting vertices.
3. Output the resulting mesh $\mathcal{M}$ = **(V, E)** as the final approximation to the input point cloud $\mathcal{P}$.

Figures 2 to 4 show the meshes reconstructed from given point clouds using our MPA algorithm.

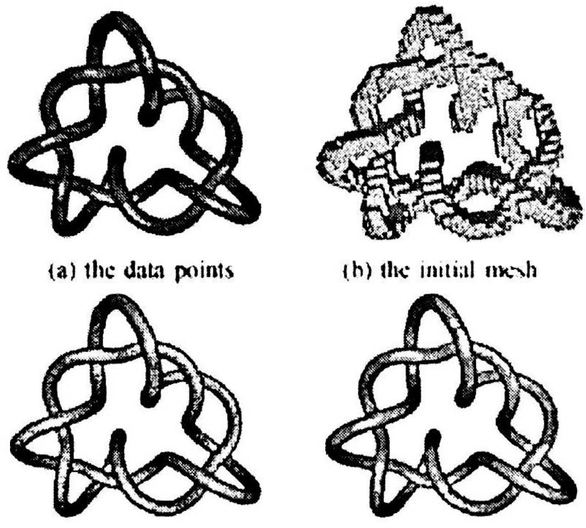
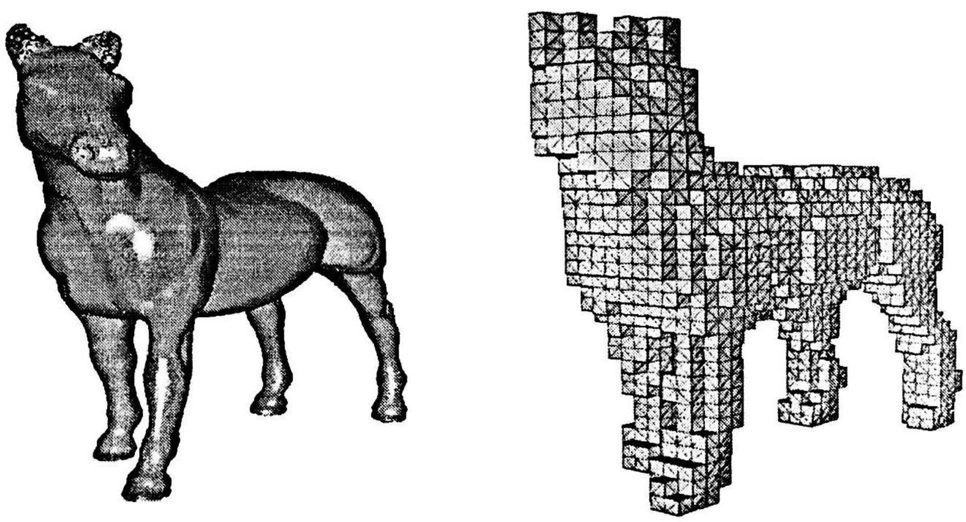
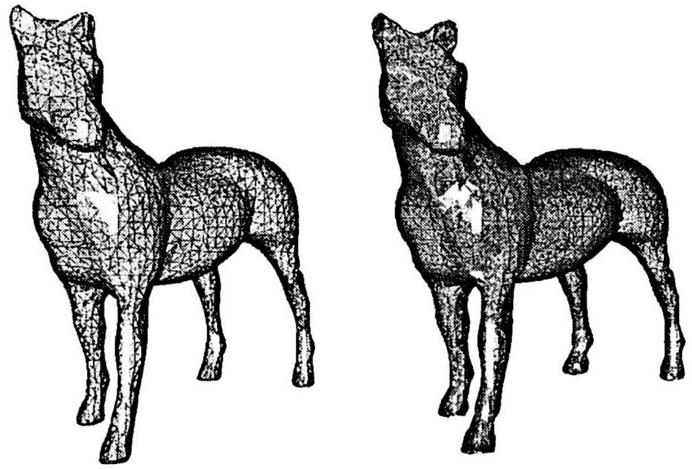


Fig. 2. Mesh reconstruction for the Knot model

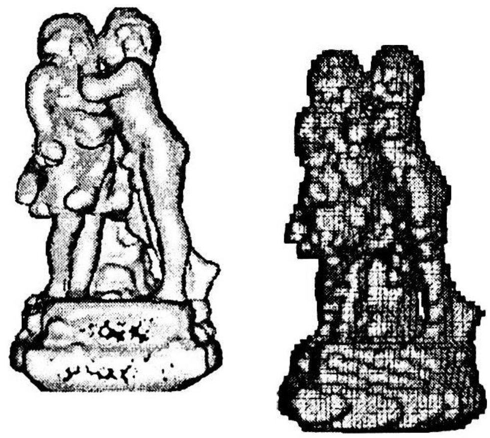


(a) the data points (b) the initial mesh

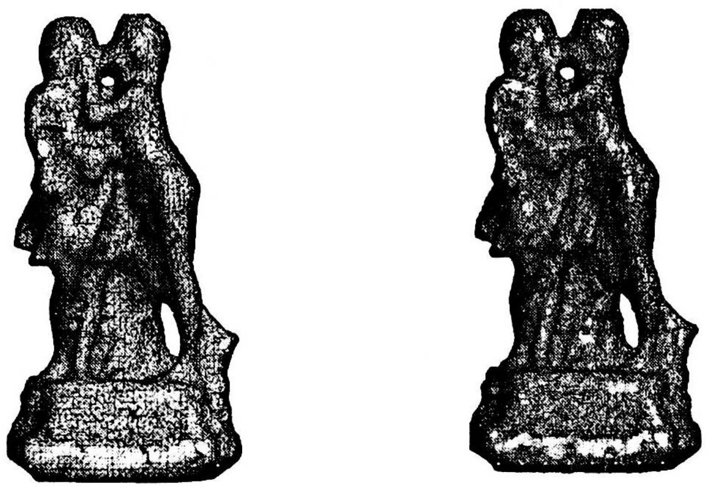


(c) the mesh after one iteration (d) the mesh after two iterations

Fig. 3. Mesh reconstruction for the Horse model.



(a) the data points (b) the initial mesh



(c) the mesh after one iteration (d) the mesh after two itenrations

Fig. 4. Mesh reconstruction for the Sculpture model.

4. Conclusion

We have shown how to construct an improved point-based representation from a point cloud, at the same time as computing the normals and curvatures of the underlying shape. Our algorithm is based on optimization theory and works robustly in the presence of noise, while yielding accurate estimates for clean data. The effectiveness of the algorithm has been demonstrated in the reconstruction of point clouds obtained by sampling several different surfaces, including a sphere, a cylinder, a paraboloid and a hyperboloid.

As an application, we use the MPA algorithm to construct a triangular mesh approximating the underlying surface of a given point cloud. We expect that our MPA method will find further applications in many operations on point-based surfaces, such as smoothing, simplification, segmentation, feature extraction, global registration.

Acknowledgments. This work was supported by grant No. R01-2005-000-11257-0 from the Basic Research Program of the Korea Science and Engineering Foundation, and in part by Seoul R&BD Program. We would like to thank the INUS Technology Inc for providing scanning data points of the Mouse model.

References

[1] A. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, C. Silva, "Point set surfaces', In *Proceedings of IEEE Visualization* (2001) 21,.

[2] D. Levin, "Mesh-independent surface interpolation", In Brunnett, B. Hamann, and H. Mueller, editors, *Geometric Modeling for Scientific Visualization*, Springer-Verlag, (2003) 37.

[3] N. Amenta, Y.J. Kil, "Defining point-set surfaces", In *Proceedings of ACM SIGGRAPH* (2004) 264.

[4] M. Pauly, R. Keiser, L.P. Kobbelt, M. Gross, "Shape modeling with point-sampled geometry", In *Proceedings of ACM SIGGRAPH* (2003) 6.

[5] P.M. do Carmo, "*Differential Geometry of Curves and Surfaces*", Prentice-Hall, 1987.

[6] R. Fletcher, "*Practical Methods of Optimization*", John Wiley & Sons, 2nd edition, 1987.

[7] M. Meyer, M. Desbrun, P. Schroder, A.H. Barr, "Discrete differential-geometry operators for triangulated 2-manifolds", In *H.C. Hege and K. Polthier, editors, Visualization and Mathematics III*, Springer-Verlag, (2003) 35.