ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8. Available from http://www.i6doc.com/en/.

Differentiable piecewise-Bézier interpolation on Riemannian manifolds

P.-A. Absil[1], Pierre-Yves Gousenbourger[1],
Paul Striewski[2] and Benedikt Wirth[2] *

1- Université catholique de Louvain - ICTEAM Institute
B-1348 Louvain-la-Neuve, Belgium

2- University of Münster - Institute for Numerical and Applied Mathematics
Einsteinstraße 62, D-48149 Münster, Germany

Abstract. We propose a generalization of classical Euclidean piecewise-Bézier surfaces to manifolds, and we use this generalization to compute a $\mathcal{C}^1$-surface interpolating a given set of manifold-valued data points associated to a regular 2D grid. We then propose an efficient algorithm to compute the control points defining the surface based on the Euclidean concept of natural $\mathcal{C}^2$-splines and show examples on different manifolds.

Fig. 1: $\mathcal{C}^1$-Bézier spline surface on the Riemannian space of shells interpolating the red shapes. The Bézier surface (gray shapes) is driven by the control points (green).

1 Introduction

This paper concerns univariate and bivariate manifold-valued interpolation with emphasis on the latter. Specifically, given data points p_{ij} in a manifold $\mathcal{M}$

*Acknowledgements: the paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme initiated by the Belgian Science Policy Office and of the DFG-funded Cells-in-Motion Cluster of Excellence (EXC1003 – CiM), University of Münster, Germany. B.W.'s research was supported by the Alfried Krupp Prize for Young University Teachers awarded by the Alfried Krupp von Bohlen und Halbach-Stiftung.

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8. Available from http://www.i6doc.com/en/.

associated to nodes $(i, j) \in \mathbb{Z}^2$ of a Cartesian grid in $\mathbb{R}^2$, we seek a $\mathcal{C}^1$ function $\mathfrak{B} : \mathbb{R}^2 \to \mathcal{M}$ such that $\mathfrak{B}(i, j) = p_{ij}$.

Several applications motivate this problem, such as projection-based model order reduction of a dynamical system depending on few parameters (where $\mathcal{M}$ is a Grassmann manifold) [1] or upsampling of diffusion tensor images (where $\mathcal{M}$ is the manifold of positive definite matrices) [2].

In contrast with the univariate case, multivariate manifold-valued interpolation does not appear much in the literature (see [3] and references therein). Steinke *et al.* [4, 5] use a thin-plate-spline technique to produce an interpolation map between two Riemannian manifolds. We also mention a related technique for volumetric registration presented in [6]. When $\mathcal{M} = \mathbb{R}^r$, on the other hand, there is a wealth of methods, in particular those based on Bézier splines [7].

In this work, we interpolate the data points by means of $\mathcal{C}^1$ piecewise-cubic Bézier surfaces (see Figure 1 for an example). First, we recall a bivariate extension [3] of manifold-valued Bézier curves [8, 9, 10]. We also give a condition to match two Bézier patches $\mathcal{C}^0$-continuously and then present a slight modification of the Bézier surface definition to ensure $\mathcal{C}^1$-continuity (Section 2). In Section 3 we provide a technique to generate Bézier control points for interpolation which is faster than in [3], and we present numerical examples in Section 4.

2 Reminder on piecewise-Bézier curves and surfaces

Bézier curves and surfaces of degree $K \in \mathbb{N}$ are functions of the form

$$\begin{aligned} \beta_K(\cdot; b_0, \ldots, b_K) : [0,1] \to \mathbb{R}^r, \qquad & t \mapsto \textstyle\sum_{j=0}^K b_j B_{jK}(t), \\ \beta_K(\cdot, \cdot; (b_{ij})_{i,j=0,\ldots,K}) : [0,1]^2 \to \mathbb{R}^r, \quad & (t_1, t_2) \mapsto \textstyle\sum_{i,j=0}^K b_{ij} B_{iK}(t_1) B_{jK}(t_2), \end{aligned}$$

where $B_{jK}(t) = \binom{K}{j} t^j (1-t)^{K-j}$ are Bernstein polynomials. They are parameterized by *control points* $b_0, \ldots, b_K \in \mathbb{R}^r$ (resp. $(b_{ij})_{i,j=0,\ldots,K} \subset \mathbb{R}^r$) which indicate the rough shape of the curve or surface and which are interpolated when their indices are in $\{0, K\}$.

Since Bernstein polynomials form a partition of unity, Bézier functions are actually convex combinations of their control points. Introducing the weighted average $\mathrm{av}[(y_1, \ldots, y_n), (w_1, \ldots, w_n)] = \mathrm{argmin}_y \sum_{i=1}^n w_i \mathrm{d}^2(y_i, y)$ with Euclidean distance d, an equivalent definition of the functions is

$$\begin{aligned} \beta_K(t; b_0, \ldots, b_K) &= \mathrm{av}[(b_i)_{i=0,\ldots,K}, (B_{iK}(t))_{i=0,\ldots,K}], \\ \beta_K(t_1, t_2; (b_{ij})_{i,j=0,\ldots,K}) &= \mathrm{av}[(b_{ij})_{i,j=0,\ldots,K}, (B_{iK}(t_1) B_{jK}(t_2))_{i,j=0,\ldots,K}]. \end{aligned} \quad (1)$$

This definition has the advantage that it generalizes to arbitrary metric spaces. In particular, this is one way among others to define Bézier functions on a Riemannian manifold $\mathcal{M}$ [3].

Let $\mathcal{M}$ be a Riemannian manifold (the special case $\mathcal{M} = \mathbb{R}^r$ is included). A piecewise-Bézier curve is defined by patching multiple Bézier curves together as

$$\mathfrak{B} : [0, M] \to \mathcal{M}, t \mapsto \beta_K(t - m; (b_i^m)_{i=0,\ldots,K}) \text{ on } [m, m+1],$$

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8. Available from http://www.i6doc.com/en/.

for $m \in \{0, \ldots, M-1\}$, and accordingly for surfaces $\mathfrak{B} : [0, M] \times [0, N] \to \mathcal{M}$. These piecewise-Bézier curves are continuous if $b_K^{m-1} = b_0^m$, $m = 1, \ldots, M-1$. The surfaces are continuous if $b_{iK}^{m,n-1} = b_{i0}^{mn}$ for all $i \in \{0, \ldots, K\}$ and $(m, n) \in \{0, \ldots, M-1\} \times \{1, \ldots, N-1\}$, and accordingly in the other direction [3].

If we additionally desire continuous differentiability, then in Euclidean space this leads to a set of additional simple constraints on the control points [7]. For piecewise-Bézier surfaces, if we allow also indices outside $\{0, \ldots, K\}$ by setting

$$b_{-1,j}^{mn} = b_{K-1,j}^{m-1,n}, \quad b_{K+1,j}^{mn} = b_{1,j}^{m+1,n}, \quad b_{j,-1}^{mn} = b_{j,K-1}^{m,n-1}, \quad b_{j,K+1}^{mn} = b_{j,1}^{m,n+1},$$

the $\mathcal{C}^1$-conditions become $b_{i0}^{mn} = \frac{b_{i,-1}^{mn} + b_{i1}^{mn}}{2}$ and $b_{0j}^{mn} = \frac{b_{-1,j}^{mn} + b_{1j}^{mn}}{2}$ for all i, j, m, n. Unfortunately, it turns out that those constraints cannot be generalized to a Riemannian manifold $\mathcal{M}$ without leading to contradictions [3]. Therefore, to achieve a $\mathcal{C}^1$ piecewise Bézier surface in $\mathcal{M}$, one has to slightly alter the definition of a Bézier surface. Indeed, the Euclidean $\mathcal{C}^1$-conditions imply that all control points b_{i0}^{mn} and b_{0j}^{mn} can be ignored and replaced by the average of their neighbors. Thus, with $\mathcal{I} = \{-1, 1, 2, \ldots, K-1, K+1\}$, one redefines [3]

$$\beta_K(t_1, t_2; (b_{ij}^{mn})_{i,j=0,\ldots,K}) = \mathrm{av}\left[(b_{ij}^{mn})_{i,j\in\mathcal{I}}, (w_i(t_1)w_j(t_2))_{i,j\in\mathcal{I}}\right] \tag{2}$$

with weights
$$w_i(t) = \begin{cases} \frac{1}{2}B_{0K}(t) & \text{if } i = -1, \\ B_{1K}(t) + \frac{1}{2}B_{0K}(t) & \text{if } i = 1, \\ B_{iK}(t) & \text{if } i = 2, \ldots, K-2, \\ B_{K-1,K}(t) + \frac{1}{2}B_{KK}(t) & \text{if } i = K-1, \\ \frac{1}{2}B_{KK}(t) & \text{if } i = K+1. \end{cases}$$

For $\mathcal{C}^1$ piecewise Bézier surfaces in Euclidean space, (2) is equivalent to (1).

3 Efficient control point generation for 1D and 2D piecewise-cubic Bézier interpolation on manifolds

Consider data points $p_{mn} \in \mathcal{M}$, $(m, n) \in \{0, \ldots, M\} \times \{0, \ldots, N\}$, where $\mathcal{M}$ is a connected, smooth, finite-dimensional manifold, and where the points are not too far from each other so that their weighted averages are well-defined. To interpolate those with a $\mathcal{C}^1$-continuous piecewise-cubic ($K = 3$) Bézier surface $\mathfrak{B} : [0, M] \times [0, N] \to \mathcal{M}$ with $\mathfrak{B}(m, n) = p_{mn}$, we need to generate appropriate control points

$$b_{ij}^{mn} \text{ for } m, n \in \{0, \ldots, M-1\} \times \{0, \ldots, N-1\} \text{ and } i, j = 1, 2.$$

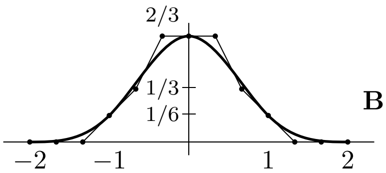
Note that in view of (2), only inner control points b_{ij}^{mn} need to be computed.

Curves. To find an appropriate method, we first consider the Euclidean space $\mathbb{R}^r$ and examine piecewise-Bézier curves. Given points p_m in $\mathbb{R}^r$, there exists a

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8. Available from http://www.i6doc.com/en/.

unique $\mathcal{C}^2$-interpolating piecewise-cubic Bézier curve $\mathfrak{B}$ whose second derivative in normal direction vanishes at the domain boundary [7, §9.3]. As a further nice characteristic, this piecewise-Bézier curve additionally minimizes the mean squared acceleration $\int_0^M \|\mathfrak{B}''(t)\|^2 \mathrm{d}t$ among all interpolating curves [7, §9.5].

Consider the B-spline representation of this optimal curve, $\mathfrak{B} = \sum_{m=-1}^{M+1} \alpha_m \mathbf{B}_m$, with coefficients $\alpha_{-1}, \dots, \alpha_{M+1} \in \mathbb{R}^r$ and with $\mathbf{B}_m = \mathbf{B}(\cdot - m)$ given by



$$\mathbf{B}(t) = \begin{cases} \beta_3(t+2; 0,0,0,\frac{1}{6}) & \text{if } t \in [-2,-1], \\ \beta_3(t+1; \frac{1}{6},\frac{1}{3},\frac{2}{3},\frac{2}{3}) & \text{if } t \in [-1,0], \\ \beta_3(t-0; \frac{2}{3},\frac{2}{3},\frac{1}{3},\frac{1}{6}) & \text{if } t \in [0,1], \\ \beta_3(t-1; \frac{1}{6},0,0,0) & \text{if } t \in [1,2], \\ 0 & \text{else.} \end{cases} \tag{3}$$

The constraints $\mathfrak{B}(m) = p_m$ and $\mathfrak{B}''(0) = \mathfrak{B}''(M) = 0$ result in the linear system

$$\underbrace{\frac{1}{6}\begin{pmatrix} 4 & 1 & & \\ 1 & \ddots & \ddots & \\ & \ddots & \ddots & 1 \\ & & 1 & 4 \end{pmatrix}}_{=:A^M} \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_{M-1} \end{pmatrix} = \underbrace{\begin{pmatrix} p_1 - \frac{p_0}{6} \\ p_2 \\ \vdots \\ p_{M-2} \\ p_{M-1} - \frac{p_M}{6} \end{pmatrix}}_{=:P^M(p_0,\dots,p_M)}, \qquad \begin{aligned} \alpha_0 &= p_0\,, \\ \alpha_M &= p_M\,, \\ \alpha_{-1} &= 2\alpha_0 - \alpha_1\,, \\ \alpha_{M+1} &= 2\alpha_M - \alpha_{M-1}\,. \end{aligned}$$

Finally, inserting (3) into $\mathfrak{B} = \sum_{m=-1}^{M+1} \alpha_m \mathbf{B}(\cdot - m)$ we see that the Bézier control points b_j^m can be computed as

$$b_0^m = p_m, \quad b_1^m = \tfrac{2}{3}\alpha_m + \tfrac{1}{3}\alpha_{m+1}, \quad b_2^m = \tfrac{1}{3}\alpha_m + \tfrac{2}{3}\alpha_{m+1}, \quad b_3^m = p_{m+1}\,.$$

Surfaces. Consider now the optimal interpolating piecewise-cubic Bézier surface $\mathfrak{B}$, still in $\mathbb{R}^r$. Its B-spline representation is $\mathfrak{B} = \sum_{m=-1}^{M+1} \sum_{n=-1}^{N+1} \alpha_{mn} \mathbf{B}_{mn}$ with $\mathbf{B}_{mn}(t_1,t_2) = \mathbf{B}_m(t_1)\mathbf{B}_n(t_2)$. Since those basis elements are just tensorised versions of the univariate case, a natural way to find the coefficients $\alpha_{mn} \in \mathbb{R}^r$ is to first identify the coefficients of the $N+1$ spline curves interpolating $p_{0n}, \dots, p_{Mn}$, $n = 0, \dots, N$, and then interpret those coefficients as interpolation points for spline curves along the other dimension. In detail, the problem to solve is now

$$\begin{aligned} &\tilde{\alpha}_{0n} = p_{0n}, \quad \tilde{\alpha}_{Mn} = p_{Mn}, \quad A^M(\tilde{\alpha}_{1n}, \dots, \tilde{\alpha}_{M-1,n})^T = P^M(p_{0n}, \dots, p_{Mn}) \quad \forall n, \\ &\alpha_{0n} = \tilde{\alpha}_{m0}, \quad \alpha_{Mn} = \tilde{\alpha}_{mN}, \quad A^N(\alpha_{m1}, \dots, \alpha_{m,N-1})^T = P^N(\tilde{\alpha}_{m0}, \dots, \tilde{\alpha}_{mN}) \quad \forall m. \end{aligned}$$

An equivalent method is the following: first, compute intermediate points

$$\tilde{p}_{mn} = \mathbf{P}(p,m,n) = P_m^M\left(P_n^N(p_{00}, \dots, p_{0N}), \dots, P_n^N(p_{M0}, \dots, p_{MN})\right) \tag{4}$$

for all (m,n); then, denoting $\bar{A} = A^{-1}$, the α_{mn} are given by

$$\alpha_{mn} = \mathbf{A}(\tilde{p},m,n) = \sum_{i=1}^{M} \sum_{j=1}^{N} \bar{A}_{mi}^M \bar{A}_{nj}^N \tilde{p}_{ij}. \tag{5}$$

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8. Available from http://www.i6doc.com/en/.

Note that the entries of $\bar{A}^M$ and $\bar{A}^N$ decay exponentially away from the diagonal. Choosing a small $d \in \mathbb{N}$ and allowing a small error, the optimal coefficients are thus approximated as $\alpha_{mn} = {}'\sum_{i=m-d}^{m+d} {}'\sum_{j=n-d}^{n+d} \bar{A}^M_{mi} \bar{A}^N_{nj} \tilde{p}_{ij}$, where ${}'\sum$ is a summation restricted to indices for which the summands are defined.

Finally, the Bézier control points b^{mn}_{ij} for $i, j \in \{1, 2\}$ are obtained via

$$b^{mn}_{ij} = \tfrac{3-i}{3}\tfrac{3-j}{3}\alpha_{mn} + \tfrac{3-i}{3}\tfrac{j}{3}\alpha_{m,n+1} + \tfrac{i}{3}\tfrac{3-j}{3}\alpha_{m+1,n} + \tfrac{i}{3}\tfrac{j}{3}\alpha_{m+1,n+1}.$$

Manifold setting. To generalize the approach to a Riemannian manifold $\mathcal{M}$, we observe that the equations stay valid under translations, that is, if we replace all α_{mn} and p_{mn} by respectively $\hat{\alpha}_{mn} = \alpha_{mn} - p_{\mathrm{ref}}$ and $\hat{p}_{mn} = p_{mn} - p_{\mathrm{ref}}$. In summary, we compute $\bar{p}_{mn} = \mathbf{P}(\hat{p}, m, n)$ and then obtain $\hat{\alpha}_{mn} = \mathbf{A}(\bar{p}, m, n)$.

On a Riemannian manifold $\mathcal{M}$, we interpret the Euclidean difference $a - p_{\mathrm{ref}}$ as a "projection" of a on the tangent space at p_{ref}. Namely, we replace all differences by logarithms $\log_{p_{\mathrm{ref}}} a$. In the computation of $\hat{\alpha}_{mn} = \log_{p_{\mathrm{ref}}} \alpha_{mn}$ one should choose $p_{\mathrm{ref}} = p_{mn}$ as the closest interpolation point. The choice of a small d now has the advantage that the computation requires only few logarithms $\log_{p_{\mathrm{ref}}} p_{\tilde{m}\tilde{n}}$ which are typically expensive to obtain and form the numerical bottleneck of the approach. At the end, $\alpha_{mn} \in \mathcal{M}$ is retrieved as $\alpha_{mn} = \exp_{p_{mn}} \hat{\alpha}_{mn}$ and the control points for $i, j \in \{1, 2\}$ as

$$b^{mn}_{ij} = \mathrm{av}[(\alpha_{mn}, \alpha_{m,n+1}, \alpha_{m+1,n}, \alpha_{m+1,n+1}), (\tfrac{3-i}{3}\tfrac{3-j}{3}, \tfrac{3-i}{3}\tfrac{j}{3}, \tfrac{i}{3}\tfrac{3-j}{3}, \tfrac{i}{3}\tfrac{j}{3})].$$

4 Numerical examples

We present here some examples of piecewise-Bézier surfaces computed on the sphere, the orthogonal group and the space of shells, with $d = 1$.

Figure 2a shows a result on $\mathbb{S}^2$ where well-known explicit formulas for logarithm and exponential map exist [11]. The $\mathcal{C}^1$-continuity of the interpolation ensures that a smooth planar curve induces a smooth curve on the surface spline. Figure 2b displays a piecewise-cubic Bézier surface in SO(3) interpolating a random set of rotations (red). Here, too, logarithm and exponential map are known explicitly, see e. g. [12]. Finally, Figure 1 is an application of the outlined method to a more complicated manifold, the space of discrete shells [13]. The Riemannian operators are in this case approximated numerically using the discrete geodesic calculus as described in [13].

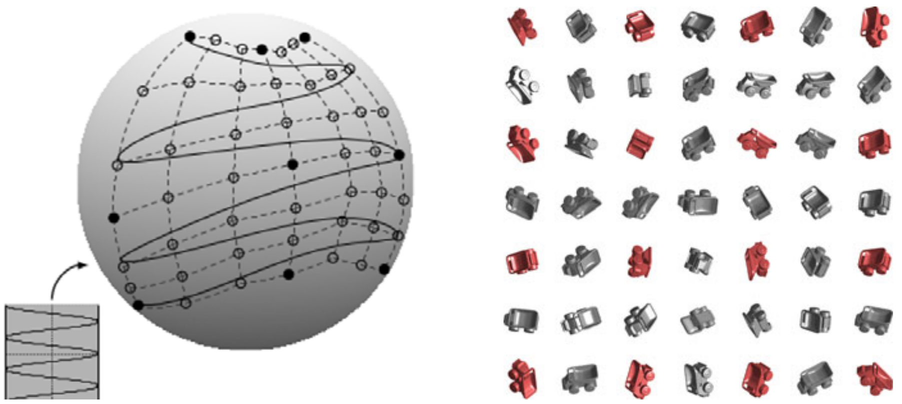
References

[1] Lorenz Pyta and Dirk Abel. Model based control of the incompressible Navier-Stokes-equations using interpolatory model reduction, 2015. To appear in the proceedings of the 54th IEEE Conference on Decision and Control.

[2] Xavier Pennec, Pierre Fillard, and Nicholas Ayache. A Riemannian framework for tensor computing. *International Journal of Computer Vision*, 66(1):41–66, 2006.

[3] P.-A. Absil, Pierre-Yves Gousenbourger, Paul Striewski, and Benedikt Wirth. Differentiable piecewise-Bézier surfaces on Riemannian manifolds. Technical report UCL-INMA-2015.10-v1, Université catholique de Louvain, 2015.

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8.
Available from http://www.i6doc.com/en/.



(a) Bivariate interpolation on $\mathbb{S}^2$. Shown are interpolation points (dots) and generated control points (circles).

(b) Bivariate interpolation on SO(3), interpolation points in red.

Fig. 2: Differentiable piecewise-cubic Bézier surfaces interpolating manifold-valued data points. Necessary control points are generated by the efficient method of Section 3.

[4] F. Steinke, M. Hein, J. Peters, and B. Schölkopf. Manifold-valued thin-plate splines with applications in computer graphics. *Computer Graphics Forum*, 27(2):437–448, apr 2008.

[5] Florian Steinke, Matthias Hein, and Bernhard Schölkopf. Nonparametric regression between general Riemannian manifolds. *SIAM Journal on Imaging Sciences*, 3(3):527–563, 2010.

[6] A.A. Joshi, D.W. Shattuck, P.M. Thompson, and R.M. Leahy. Surface-constrained volumetric brain registration using harmonic mappings. *Medical Imaging, IEEE Transactions on*, 26(12):1657–1669, Dec 2007.

[7] G. Farin. *Curves and Surfaces for CAGD*. Academic Press, fifth edition, 2002.

[8] A. Lin and M. Walker. CGD techniques for differentiable manifolds. *Proceedings of the 2001 International Symposium Algorithms for Approximation IV*, 2001.

[9] T. Popiel and L. Noakes. Bézier curves and $\mathcal{C}^2$ interpolation in Riemannian manifolds. *J. Approx. Theory*, pages 148(2):111–127, 2007.

[10] A Arnould, P.-Y. Gousenbourger, C Samir, P.-A. Absil, and M Canis. Fitting Smooth Paths on Riemannian Manifolds : Endometrial Surface Reconstruction and Preoperative MRI-Based Navigation. In F.Nielsen and F.Barbaresco, editors, *GSI2015*, pages 491–498. Springer International Publishing, 2015.

[11] Q. Rentmeesters. A gradient method for geodesic data fitting on some symmetric Riemannian manifolds. In *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pages 7141–7146, 2011.

[12] Nicolas Boumal and P.-A. Absil. A discrete regression method on manifolds and its application to data on SO(n). In *IFAC Proceedings Volumes (IFAC-PapersOnline)*, volume 18, pages 2284–2289, 2011.

[13] B. Heeren, M. Rumpf, P. Schröder, M. Wardetzky, and B. Wirth. Exploring the geometry of the space of shells. *Computer Graphics Forum*, 33(5):247–256, 2014.